Vulnerability Assessment Report of

Visiview Web Application for Hillrom

Aug 2021

About L&T Technology Services:
L&T Technology Services Limited (LTTS) is a global leader in Engineering and R&D (ER&D) services. With 399 patents filed for 51 of the Global Top 100 ER&D spenders. Our innovations speak for itself – World's 1st Autonomous Welding Robot, Solar 'Connectivity' Drone, and the Smartest Campus in the World, to name a few. LTTS expertise in engineering design, product development, smart manufacturing, and digitalization touches every area of our lives. With 49 Innovation and R&D design centers globally, we specialize in disruptive technology spaces such as 5G, Artificial Intelligence, Collaborative Robots, Digital Factory, and Autonomous Transport. **LTTS is a publicly listed subsidiary of Larsen & Toubro Limited, the $18 billion Indian conglomerate operating in over 30 countries.**

Document History:

Var. Rel. No.	Release Date	Prepared By. Prepared. Dt.	Reviewed By Reviewed Dt.	Approved By Approval Dt.	Remark/Revision Details
1.0	21-08-2021	Daliya Pallavi 21-08-2021	Radha Krishna 21-08-2021	Atanu Niyogi 21-08-2021	Initial Release

Table of Contents

1. Overview of the project

L&T Technology Services (LTTS) security team has conducted Security Assessment for Visiview Web Application for Hillrom. The purpose of the assessment is to evaluate the security posture of the web application against common vulnerabilities

Objective of the security assessment:
As a part of this engagement a holistic approach was taken to conduct the Vulnerability Assessment and Penetration Testing on Visiview Web Application for Hillrom. During the engagement High, Medium, and Low severity issues were identified with respect to Hillrom web application.

Approach
The following approach was taken to make sure the target was assessed against known vulnerabilities from all possible security perspectives:

- Manual Vulnerability Assessment and Penetration Testing using OWASP TOP 10 for web application.

Some of the tools which were used are listed below:

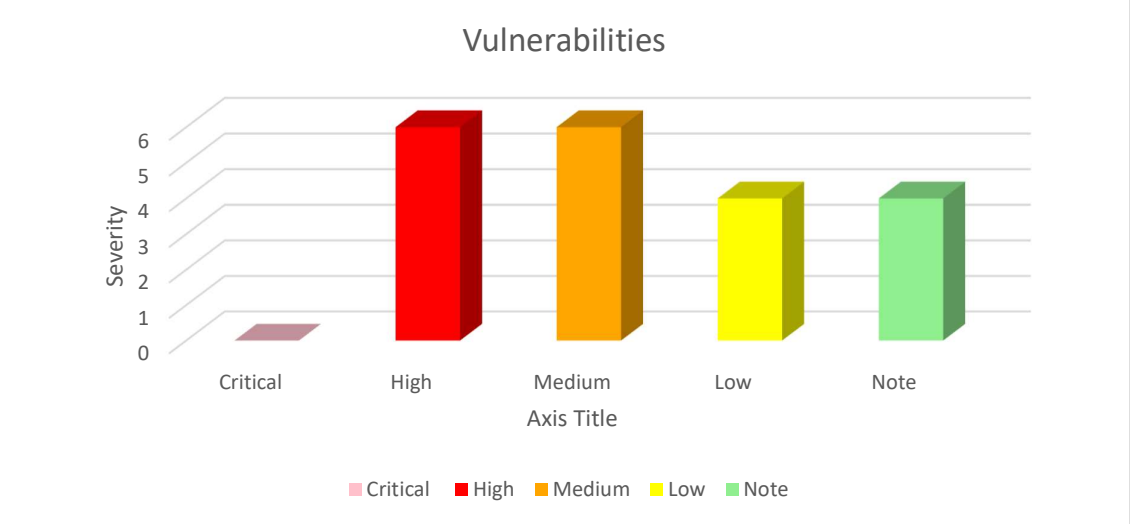
Target Application	Visiview Hillrom
Browser	Chrome, Firefox
Tools	Burp, ZAP, Wireshark, Qualys, Nmap, Nikto

Key Security Policies
OWASP top 10 listed vulnerabilities ware used as a reference framework. The following key security aspects were checked:

1. Injection
2. Broken Authentication
3. Sensitive Data Exposure
4. XML External Entities (XXE)
5. Broken Access Control
6. Security Misconfiguration
7. Cross-Site Scripting (XSS)
8. Insecure Deserialization
9. Using Components with Known Vulnerabilities
10. Insufficient Logging & Monitoring

Summary of Findings

The graph below shows a summary of the number of vulnerabilities found for each impact level for the Application Security Assessment. Vulnerabilities found are addressed according to priority, findings, analysis, and recommendations from the assessment.



Overall Risk Severity = Likelihood x Impact				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

2. Vulnerabilities explained in detail

2.1 Privilege Escalation Modifying Token			
Impact	High	**Risk Rating**	High
Ease of Exploit	Easy		
Likelihood	Medium		
Category	Reliance on Cookies without Validation and Integrity Checking		
URL/Impacted system	https://visiview.hillrom.com/		

Description

The application does not check or incorrectly checks entitlements when a user attempts to access privileged resources or actions. Given that a user has already authenticated to the application, authorization checks exist to enforce what functions that user is allowed to access. These checks may be applied inconsistently, in the wrong locations, or may not exist at all. A common example of this is an application that relies on UI controls to prevent access to administrative functionality. Although the pages containing these functions are not visible in the UI, they are still accessible if the address is known.

Impact

A malicious user may exploit a broken authorization mechanism to perform actions or access resources they should not be able to access. This may manifest as either horizontal or vertical privilege escalation depending on the nature of the resource in question. Without a functioning authorization mechanism, confidentiality, data integrity, and application availability may all be affected if the resource compromised is relevant.

Recommendation

Authorization should always be explicitly checked server-side to verify that the user making a request to view data or perform an action is authorized to do so. This may be as simple as checking that a user is authorized to view a particular page in the application or check that the user is authorized to perform an action overall. Many other more fine-grained situations may exist depending on the application context and the complexity of the business functionality. The following are some common situations where a more fine-grained authorization check is necessary:

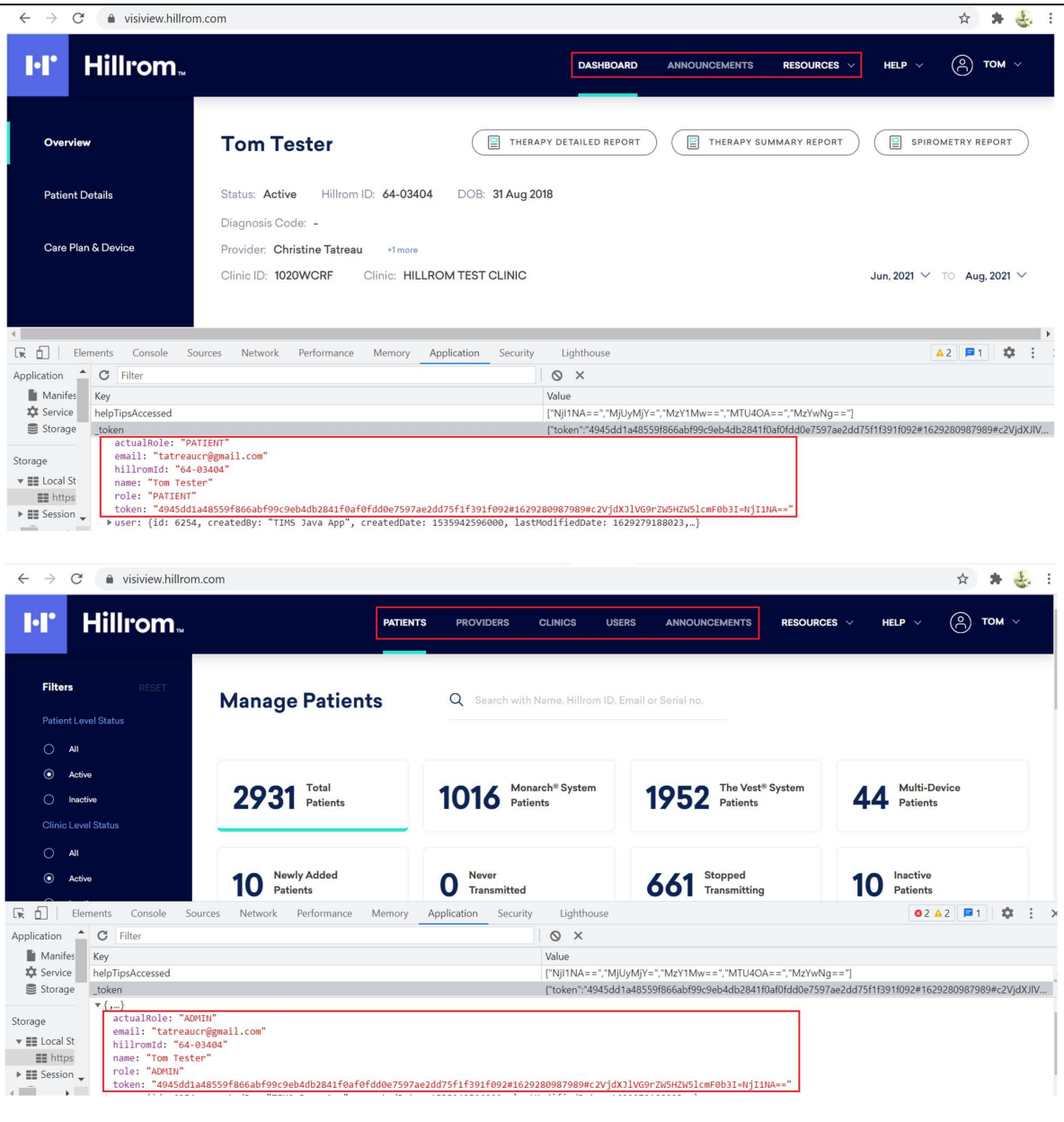
- A user may be authorized to perform an action, but only against certain entities involved in the transaction. For example, a user may be authorized to perform a funds transfer from one account to another, but they may only be authorized to access certain accounts. Authorization checks must be performed to verify that the user is authorized to make a funds transfer as well as verify that the user is authorized to perform that transaction using the supplied accounts.
- A user may be authorized to view only data tied to their own account. For example, a retail banking customer may be authorized to view the details of each of their accounts at a particular bank but may not be authorized to view the details of other users' accounts. Authorization checks must be performed to verify that the user is authorized to view

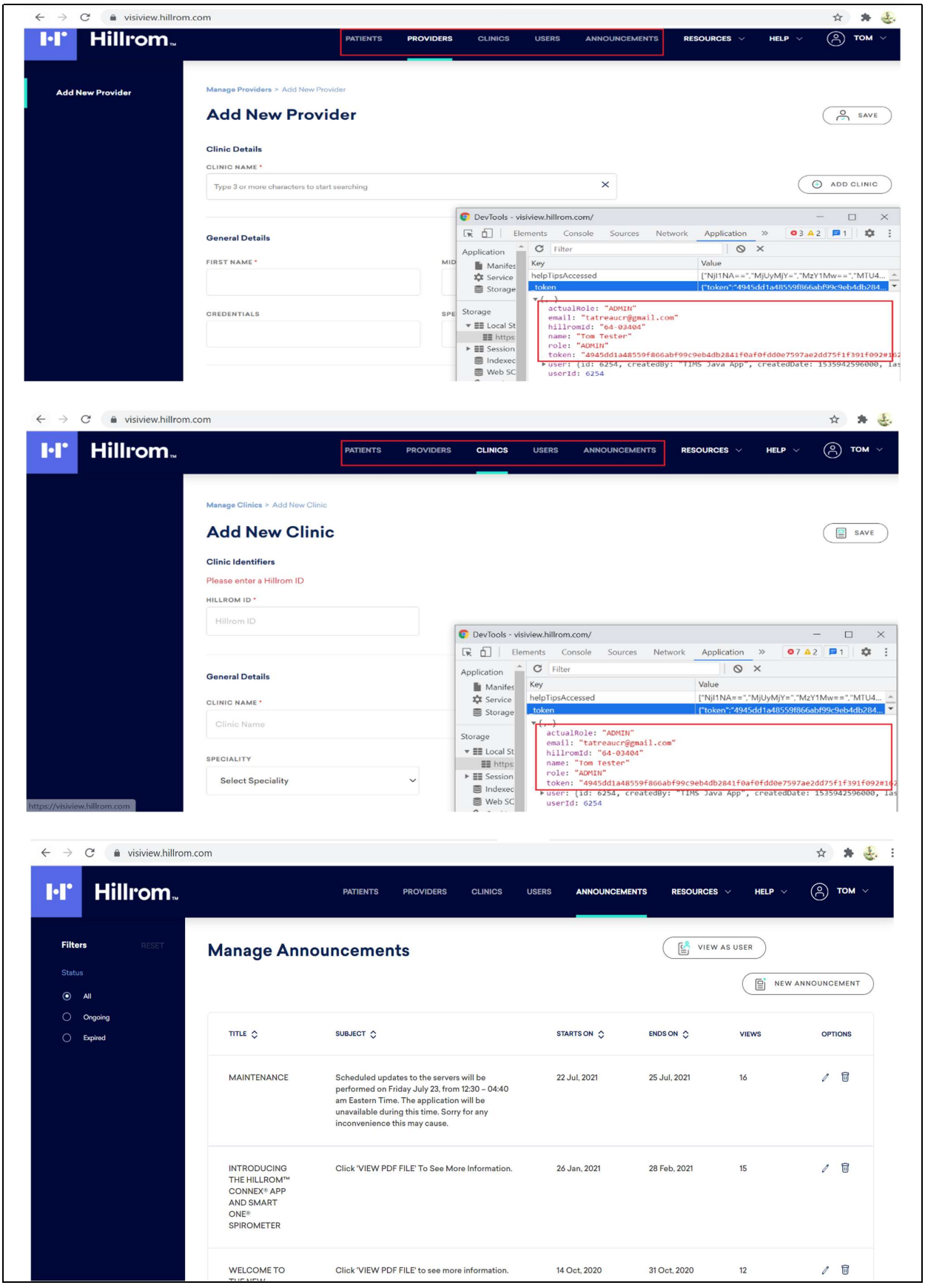
account details as well as verify that the user is authorized to view the details of the account, they have requested information for.

How to recreate the Security defect

- Login to the application using PATIENT credentials.
- Right click and inspect element. Go to local storage.
- Change the actual Role from PATIENT to ADMIN and reload the page
- We can access the properties which admin user have.

Evidence




visiview.hillrom.com
Hillrom
PATIENTS
PROVIDERS
CLINICS
USERS
ANNOUNCEMENTS
RESOURCES
HELP
TOM
Add New Provider
Manage Providers > Add New Provider
Add New Provider
SAVE
Clinic Details
CLINIC NAME *
Type 3 or more characters to start searching
ADD CLINIC
General Details
FIRST NAME *
CREDENTIALS
DevTools - visiview.hillrom.com/
Elements
Console
Sources
Network
Application
Key
Value
helpTipsAccessed
token
actualRole: "ADMIN"
email: "tatreaucr@gmail.com"
hillromId: "64-03404"
name: "Tom Tester"
role: "ADMIN"
token: "4945dd1a48559f866abf99c9eb4db2841f0af0fdd0e7597ae2dd75f1f391f092#162"
userId: 6254
visiview.hillrom.com
Hillrom
PATIENTS
PROVIDERS
CLINICS
USERS
ANNOUNCEMENTS
RESOURCES
HELP
TOM
Manage Clinics > Add New Clinic
Add New Clinic
SAVE
Clinic Identifiers
Please enter a Hillrom ID
HILLROM ID *
Hillrom ID
General Details
CLINIC NAME *
Clinic Name
SPECIALITY
Select Speciality
https://visiview.hillrom.com
DevTools - visiview.hillrom.com/
Elements
Console
Sources
Network
Application
Key
Value
helpTipsAccessed
token
actualRole: "ADMIN"
email: "tatreaucr@gmail.com"
hillromId: "64-03404"
name: "Tom Tester"
role: "ADMIN"
token: "4945dd1a48559f866abf99c9eb4db2841f0af0fdd0e7597ae2dd75f1f391f092#162"
userId: 6254
visiview.hillrom.com
Hillrom
PATIENTS
PROVIDERS
CLINICS
USERS
ANNOUNCEMENTS
RESOURCES
HELP
TOM
Filters
RESET
Status
All
Ongoing
Expired
Manage Announcements
VIEW AS USER
NEW ANNOUNCEMENT
TITLE
SUBJECT
STARTS ON
ENDS ON
VIEWS
OPTIONS
MAINTENANCE
Scheduled updates to the servers will be performed on Friday July 23, from 12:30 – 04:40 am Eastern Time. The application will be unavailable during this time. Sorry for any inconvenience this may cause.
22 Jul, 2021
25 Jul, 2021
16
INTRODUCING THE HILLROM™ CONNEX® APP AND SMART ONE® SPIROMETER
Click 'VIEW PDF FILE' To See More Information.
26 Jan, 2021
28 Feb, 2021
15
WELCOME TO
Click 'VIEW PDF FILE' to see more information.
14 Oct, 2020
31 Oct, 2020
12

2.2 Insecure Direct Object References (IDOR)			
Impact	High	**Risk Rating**	High
Ease of Exploit	Easy		
Likelihood	Medium		
Category	Improper Privilege Management		
URL/Impacted system	https://visiview.hillrom.com/		

Description

Insecure direct object references (IDOR) are a cybersecurity issue that occurs when a web application developer uses an identifier for direct access to an internal implementation object but provides no additional access control and/or authorization checks. For example, an IDOR vulnerability would happen if the URL of a transaction could be changed through client-side user input to show unauthorized data of another transaction.

Impact

Exposure of Confidential Information: When the attacker will have control over your account via this vulnerability, it is obvious that an attacker will be able to come across your personal information.

Recommendation

- Developers should avoid displaying private object references such as keys or file names.
- Validation of Parameters should be properly implemented.
- Verification of all the Referenced objects should be done.
- Tokens should be generated in such a way that it should only be mapped to the user and should not be public.

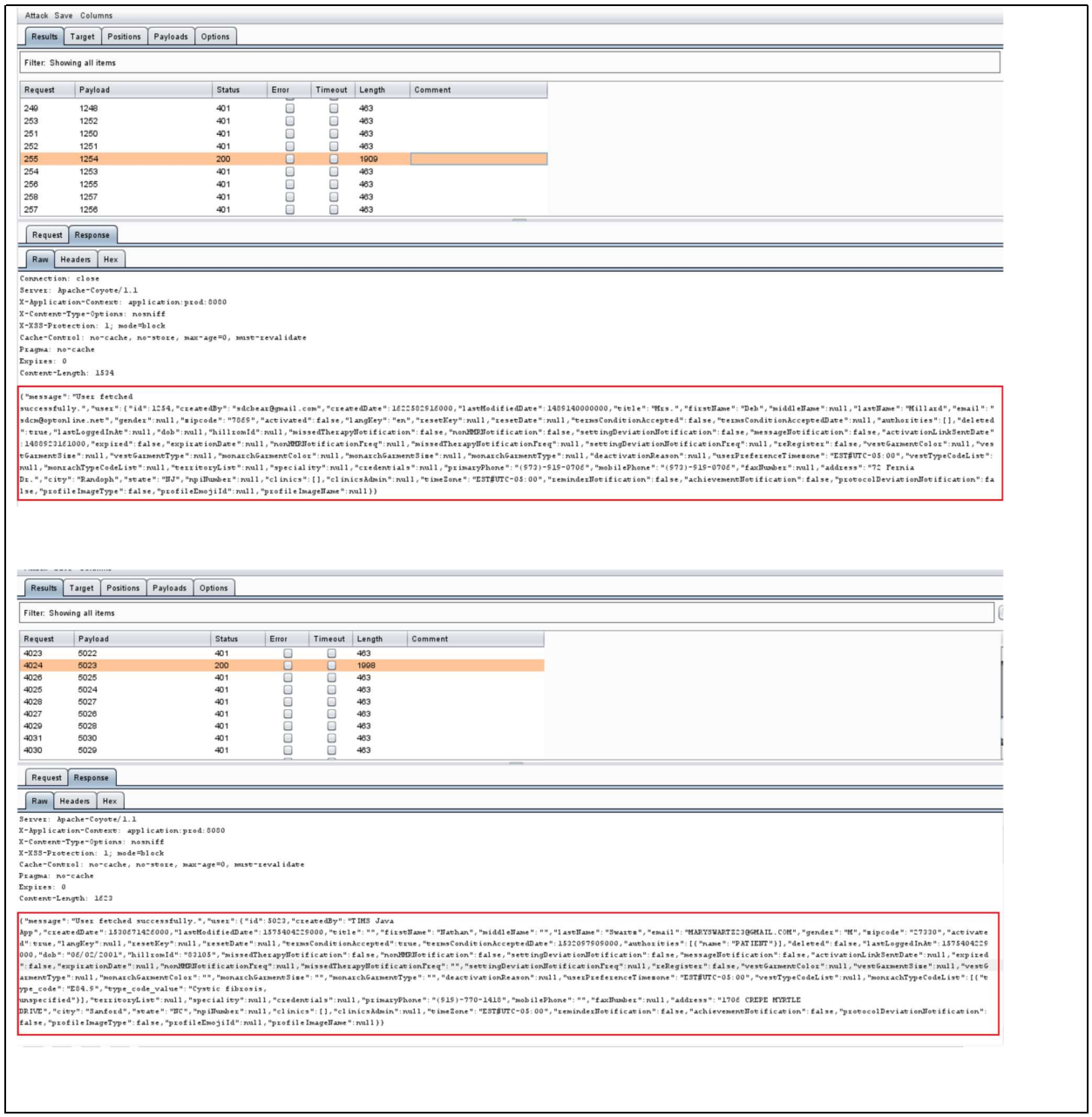
How to recreate the Security defect

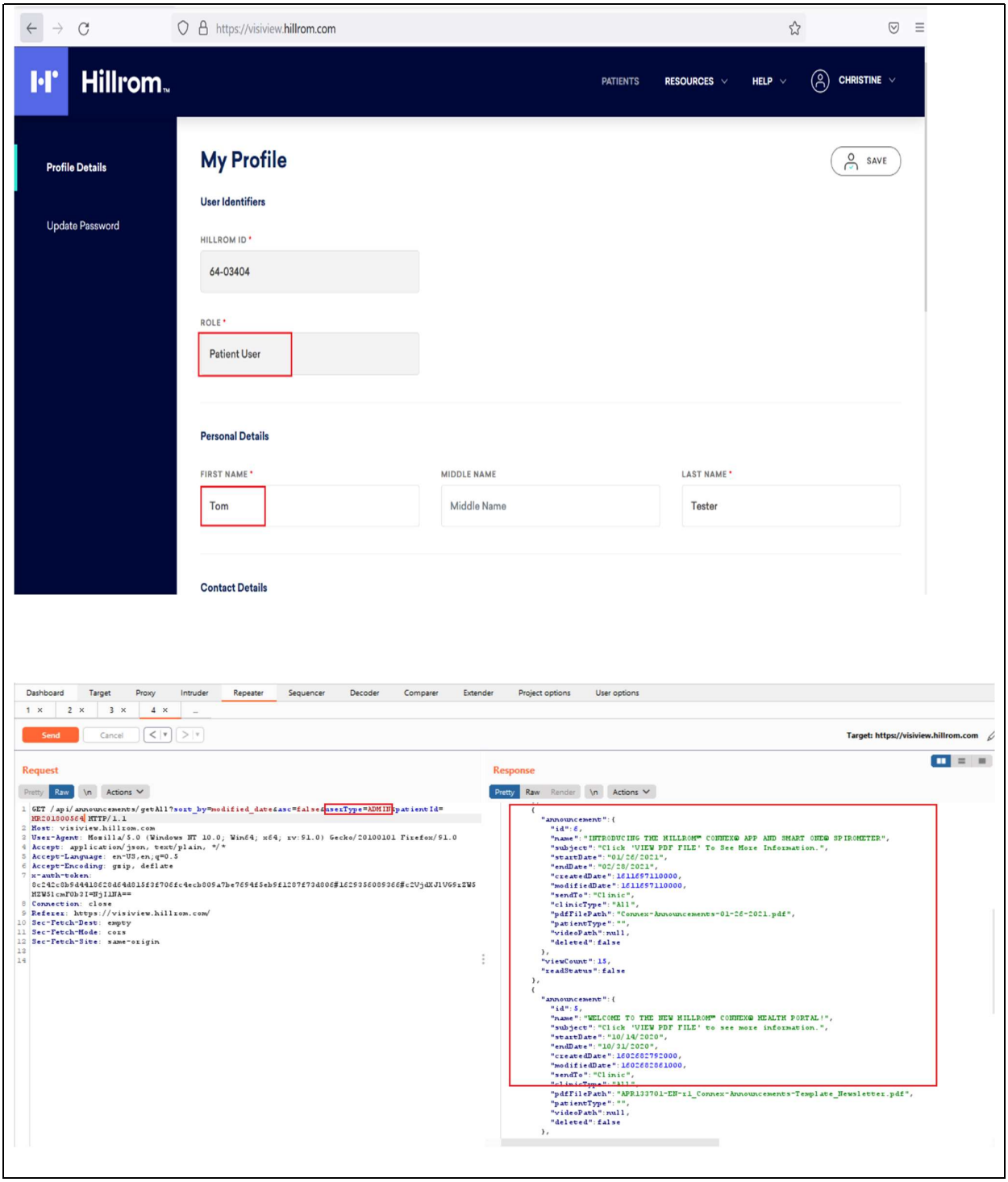
- Login to the web application.
- Capture the user profile request with burp suite.
- Forward the request to intruder.
- Give payloads (1000-4000 numbers) in the Id position
- After successful attack, we can get other user profile details.

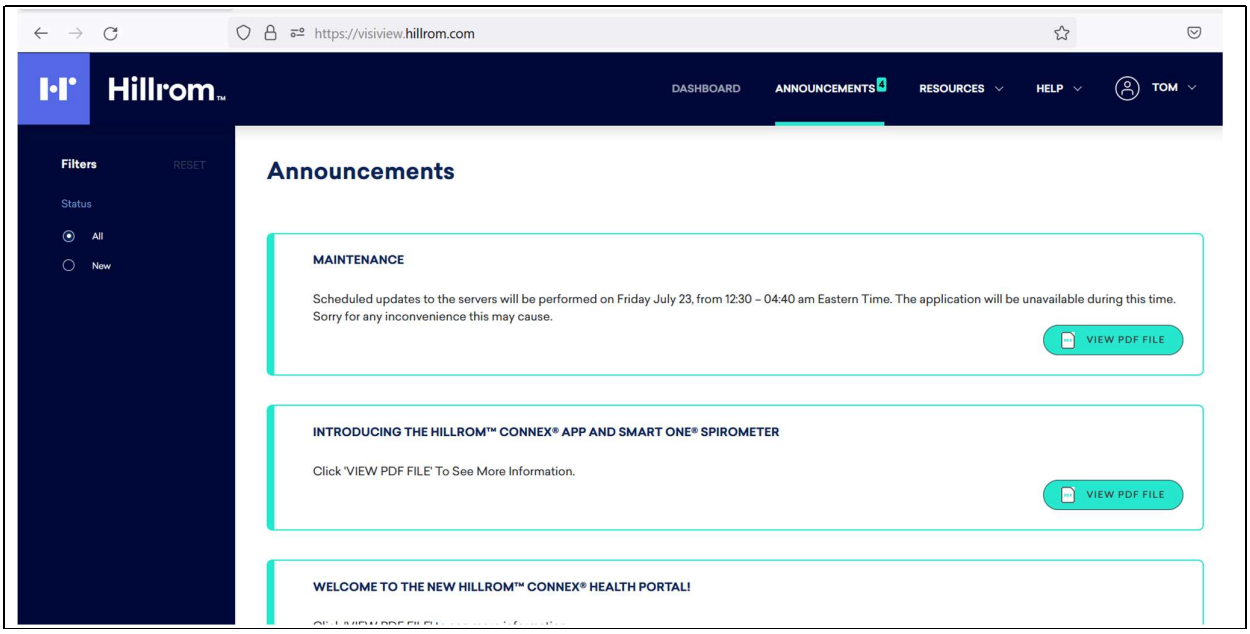
And

- Login to the web application
- Capture the announcement request with burp suite
- Change ‘user type’ form PATIENT to ADMIN
- After successful attack, we can see various announcements from admin portal

Evidence


Attack Save Columns
Results Target Positions Payloads Options
Filter: Showing all items
Request Payload Status Error Timeout Length Comment
249 1248 401 463
253 1252 401 463
251 1250 401 463
252 1251 401 463
255 1254 200 1909
254 1253 401 463
256 1255 401 463
258 1257 401 463
257 1256 401 463
Request Response
Raw Headers Hex
Connection: close
Server: Apache-Coyote/1.1
X-Application-Context: application:prod:8080
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Content-Length: 1534
{"message":"User fetched successfully.","user":{"id":1254,"createdBy":"sdchear@gmail.com","createdDate":1572582916000,"lastModifiedDate":1605140000000,"title":"Mrs.","firstName":"Deb","middleName":null,"lastName":"Millard","email":"sdcm@optonline.net","gender":null,"zipcode":"7869","activated":false,"langKey":"en","resetKey":null,"resetDate":null,"termsConditionAccepted":false,"termsConditionAcceptedDate":null,"authorities":[],"deleted":true,"lastLoggedInAt":null,"dob":null,"hillromId":null,"missedTherapyNotification":false,"nonNMKNotification":false,"settingDeviationNotification":false,"messageNotification":false,"activationLinkSentDate":1605921616000,"expired":false,"expirationDate":null,"nonNMKNotificationFreq":null,"missedTherapyNotificationFreq":null,"settingDeviationNotificationFreq":null,"reRegister":false,"vestGarmentColor":null,"vestGarmentSize":null,"vestGarmentType":null,"monarchGarmentColor":null,"monarchGarmentSize":null,"monarchGarmentType":null,"deactivationReason":null,"userPreferenceTimezone":"EST#UTC-05:00","vestTypeCodeList":null,"monarchTypeCodeList":null,"territoryList":null,"speciality":null,"credentials":null,"primaryPhone":"(972)-515-0706","mobilePhone":"(972)-515-0706","faxNumber":null,"address":"72 Fernia Dr.","city":"Randolph","state":"NJ","npiNumber":null,"clinics":[],"clinicsAdmin":null,"timeZone":"EST#UTC-05:00","reminderNotification":false,"achievementNotification":false,"protocolDeviationNotification":false,"profileImageType":false,"profileEmojiId":null,"profileImageName":null}}
Results Target Positions Payloads Options
Filter: Showing all items
Request Payload Status Error Timeout Length Comment
4023 5022 401 463
4024 5023 200 1998
4026 5025 401 463
4025 5024 401 463
4028 5027 401 463
4027 5026 401 463
4029 5028 401 463
4031 5030 401 463
4030 5029 401 463
Request Response
Raw Headers Hex
Server: Apache-Coyote/1.1
X-Application-Context: application:prod:8080
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Content-Length: 1623
{"message":"User fetched successfully.","user":{"id":5022,"createdBy":"TIMS Java App","createdDate":1590671425000,"lastModifiedDate":1575404229000,"title":"","firstName":"Nathan","middleName":"","lastName":"Swartz","email":"MARYSWARTZ23@GMAIL.COM","gender":"M","zipcode":"27330","activated":true,"langKey":null,"resetKey":null,"resetDate":null,"termsConditionAccepted":true,"termsConditionAcceptedDate":1532067505000,"authorities":[{"name":"PATIENT"}],"deleted":false,"lastLoggedInAt":1575404229000,"dob":"06/02/2001","hillromId":"93105","missedTherapyNotification":false,"nonNMKNotification":false,"settingDeviationNotification":false,"messageNotification":false,"activationLinkSentDate":null,"expired":false,"expirationDate":null,"nonNMKNotificationFreq":null,"missedTherapyNotificationFreq":"","settingDeviationNotificationFreq":null,"reRegister":false,"vestGarmentColor":null,"vestGarmentSize":null,"vestGarmentType":null,"monarchGarmentColor":"","monarchGarmentSize":"","monarchGarmentType":"","deactivationReason":null,"userPreferenceTimezone":"EST#UTC-05:00","vestTypeCodeList":null,"monarchTypeCodeList":[{"type_code":"E84.9","type_code_value":"Cystic fibrosis, unspecified"}],"territoryList":null,"speciality":null,"credentials":null,"primaryPhone":"(515)-770-1418","mobilePhone":"","faxNumber":null,"address":"1706 CREPE MYRTLE DRIVE","city":"Sanford","state":"NC","npiNumber":null,"clinics":[],"clinicsAdmin":null,"timeZone":"EST#UTC-05:00","reminderNotification":false,"achievementNotification":false,"protocolDeviationNotification":false,"profileImageType":false,"profileEmojiId":null,"profileImageName":null}}


https://visiview.hillrom.com
Hillrom
PATIENTS
RESOURCES
HELP
CHRISTINE
Profile Details
Update Password
My Profile
SAVE
User Identifiers
HILLROM ID *
64-03404
ROLE *
Patient User
Personal Details
FIRST NAME *
Tom
MIDDLE NAME
Middle Name
LAST NAME *
Tester
Contact Details
Dashboard
Target
Proxy
Intruder
Repeater
Sequencer
Decoder
Comparer
Extender
Project options
User options
Send
Cancel
Target: https://visiview.hillrom.com
Request
Response
GET /api/announcements/getAll?sort_by=modified_date&asc=false&userType=ADMIN&patientId=
Host: visiview.hillrom.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
x-auth-token:
Connection: close
Referer: https://visiview.hillrom.com/
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
"announcement":{
"id":6,
"name":"INTRODUCING THE HILLROM™ CONNEX® APP AND SMART ONE® SPIROMETER",
"subject":"Click 'VIEW PDF FILE' To See More Information.",
"startDate":"01/26/2021",
"endDate":"02/28/2021",
"createdDate":1611697110000,
"modifiedDate":1611697110000,
"sendTo":"Clinic",
"clinicType":"All",
"pdfFilePath":"Connex-Announcements-01-26-2021.pdf",
"patientType":"",
"videoPath":null,
"deleted":false
},
"viewCount":15,
"readStatus":false
},
{
"announcement":{
"id":5,
"name":"WELCOME TO THE NEW HILLROM™ CONNEX® HEALTH PORTAL!",
"subject":"Click 'VIEW PDF FILE' to see more information.",
"startDate":"10/14/2020",
"endDate":"10/31/2020",
"createdDate":1602682792000,
"modifiedDate":1602682861000,
"sendTo":"Clinic",
"clinicType":"All",
"pdfFilePath":"APR133701-EN-xl_Connex-Announcements-Template_Newsletter.pdf",
"patientType":"",
"videoPath":null,
"deleted":false
},



2.3 Application Accessible over HTTP			
Impact	High	**Risk Rating**	High
Ease of Exploit	Easy		
Likelihood	Medium		
Category	Cleartext Transmission of Sensitive Information		
URL/Impacted system	http://visiview.hillrom.com/		

Description

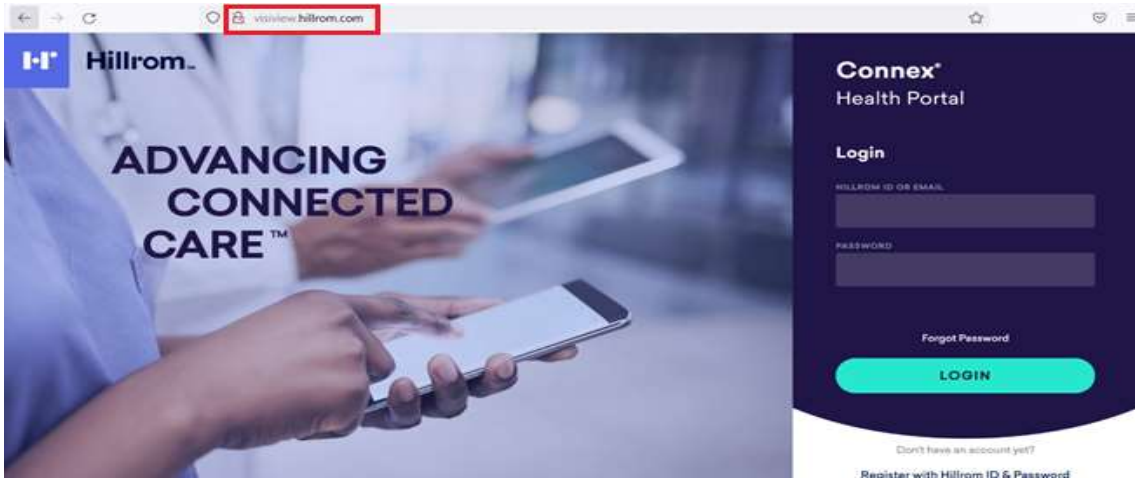
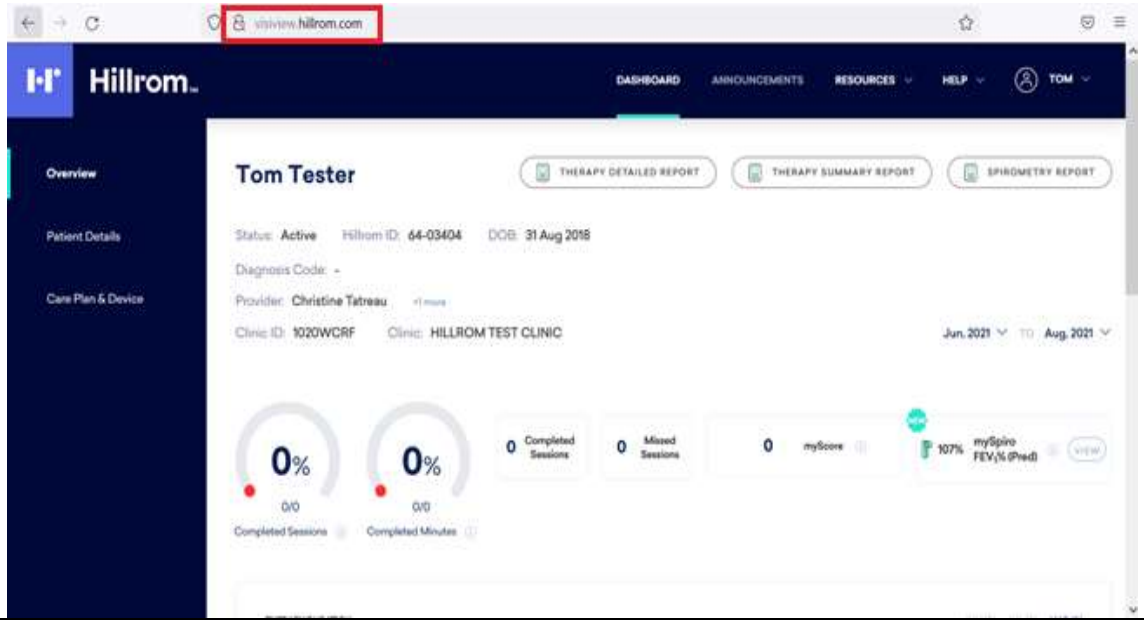
The Application allows web browsers to access to the application over HTTP and doesn't redirect them to HTTPS. The application fails to prevent users from connecting to it over unencrypted connections.

Impact

- This is performed by rewriting HTTPS links as HTTP, so that if a targeted user follows a link to the site from an HTTP page, their browser never attempts to use an encrypted connection.
- An attacker able to modify a legitimate user's network traffic could bypass the application's use of SSL/TLS encryption and use the application as a platform for attacks against its users.

Recommendation

- The application should instruct web browsers to only access the application using HTTPS.
- Enable HTTP Strict Transport Security (HSTS) by adding a response header with the name 'Strict-Transport-Security' and the value 'max-age=expireTime', where expireTime is the time in seconds that browsers should remember that the site should only be accessed.

How to recreate the Security defect
• Browse to – http://visiview.hillrom.com/ • Application does not redirect to HTTPS and the traffic sent over HTTP.
Evidence
   

2.4 Improper Error Handling			
Impact	High	**Risk Rating**	High
Ease of Exploit	Easy		
Likelihood	Medium		
Category	Observable Response Discrepancy		

URL/Impacted system	https://visiview.hillrom.com/

Description

The application's login functionality returns different responses depending on whether the entered username is valid or not. The difference in responses may be as straightforward as "An email with password reset instructions was sent to the user's email address on file" versus "User [username] does not exist". Responses may also be subtler. An attacker can abuse this design to compile a list of valid users through automated brute force guessing attempts.

Impact

Username enumeration provides an attacker with one of two pieces of information required to authenticate to the application. By automating guesses, an attacker can retrieve a large list of valid usernames for an application. Once the attacker has a list of valid usernames, they can begin guessing passwords to steal credentials and impersonate other users. Password guessing attempts may be done manually, or via automated means depending on what login anti-automation mechanisms (if any) the application has in place. Valid usernames may also be used in phishing exercises as well as large-scale account lockout denial of service attacks.

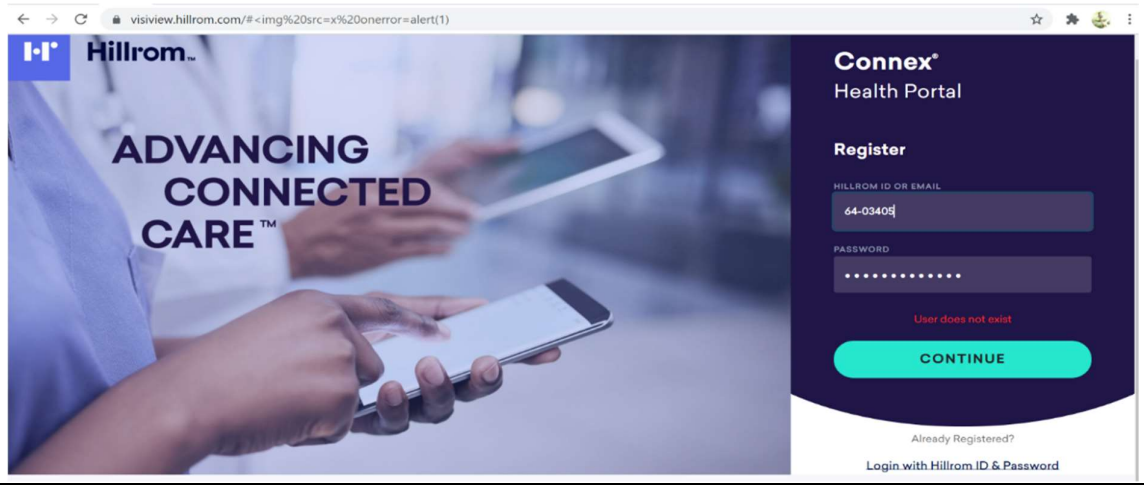
Recommendation

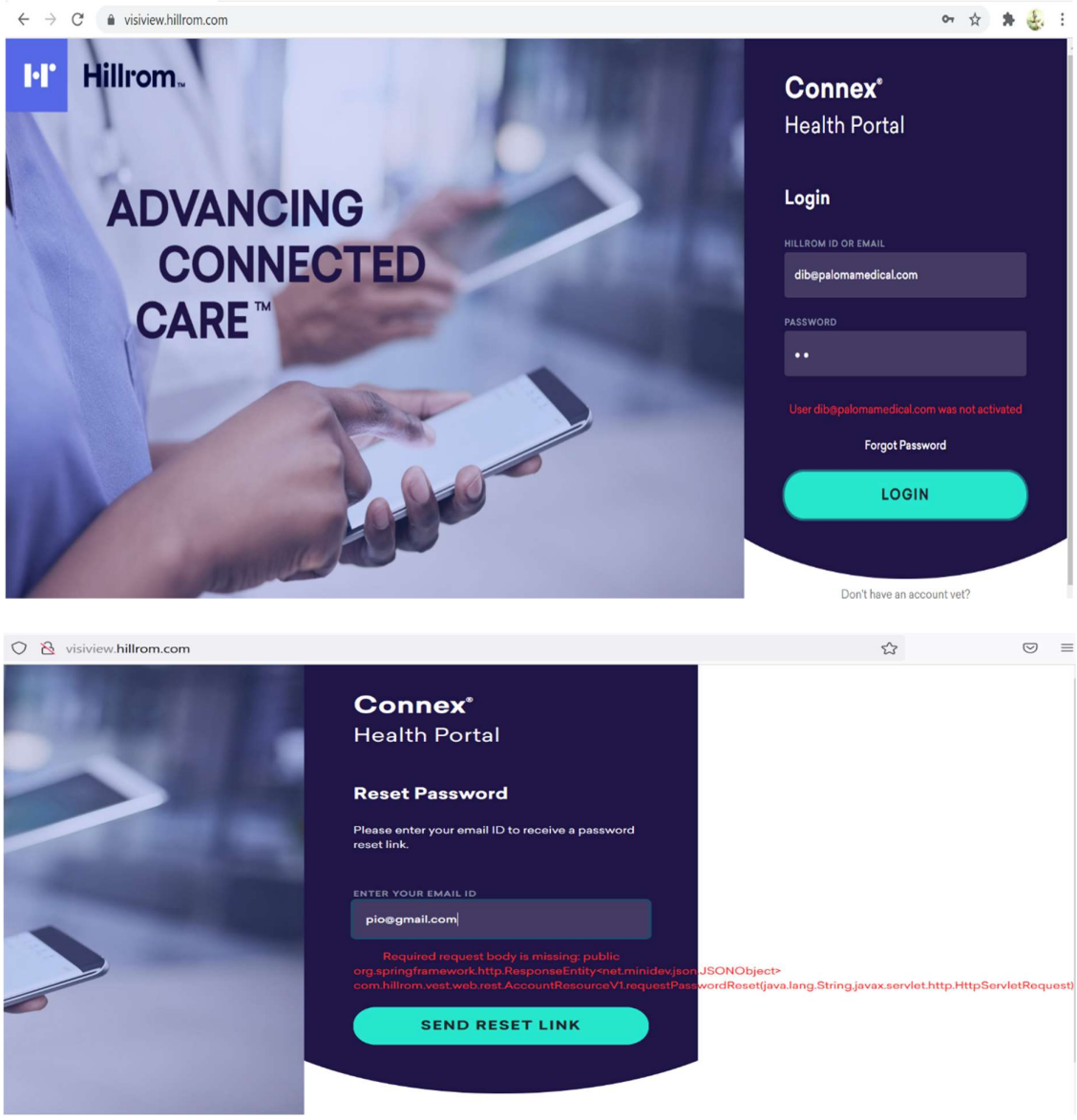
- The application should return the same response whether or not the supplied username is associated with a valid account, e.g., "If the supplied username does not exist or supplied password is incorrect, make sure to return a generic “No such username or password” message. This prevents users from guessing valid usernames based on the server's response time.
- Make sure the HTTP response, and the time taken to respond are no different when a username does not exist, and an incorrect password is entered.

How to recreate the Security defect

- Browse to – https://visiview.hillrom.com/
- Enter invalid username and password
- Click on the login button

Evidence





2.5 No Account Lockout Policy			
Impact	High	**Risk Rating**	High
Ease of Exploit	Moderate		
Likelihood	Medium		
Category	Improper Restriction of Excessive Authentication Attempts		
URL/Impacted system	https://visiview.hillrom.com/		
Description			
Current account lockout mechanism can mitigate brute force password guessing attacks. Accounts are typically locked after 3 to 5 unsuccessful login attempts and can only be unlocked after a			

predetermined period, via a self-service unlock mechanism, or intervention by an administrator. Account lockout mechanisms require a balance between protecting accounts from unauthorized access and protecting users from being denied authorized access.

Impact

It is possible for a malicious user to gain access to the application by brute forcing the password. Since there are no restrictions on the number of logins attempts a malicious user can brute force the credentials of a user, until the right credentials are guessed.

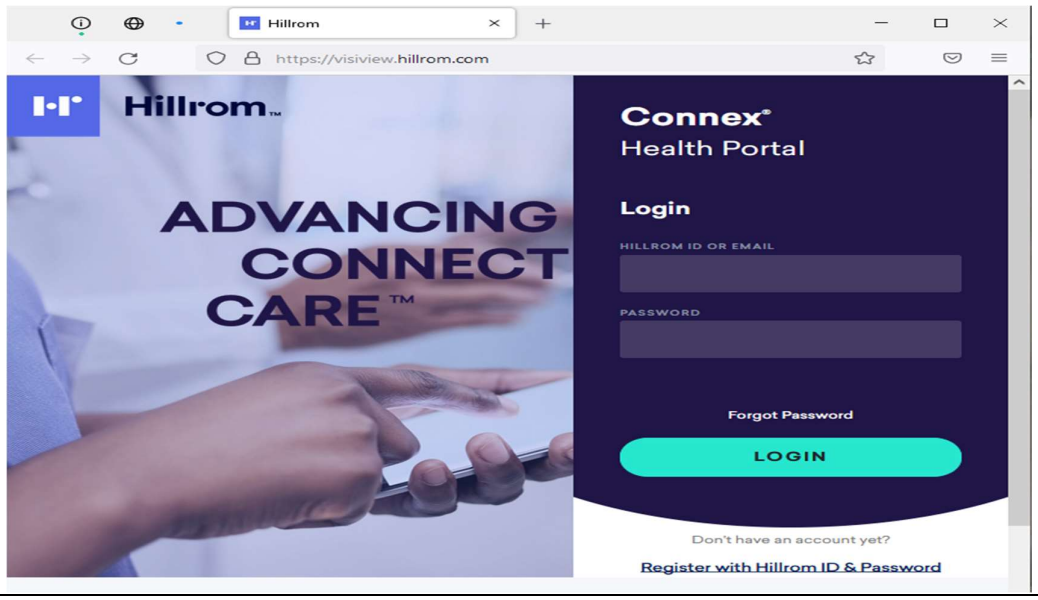
Recommendation

- The most obvious way to block brute-force attacks is to simply lock out accounts after a defined number of incorrect password attempts. Account lockouts can last a specific duration, such as one hour, or the accounts could remain locked until manually unlocked by an administrator. A CAPTCHA may hinder brute force attacks, but CAPTCHA should be perceived as a rate limiting protection only which stops the attacker for a limited amount of time, also can use alternative verification channels like multi factor authentication.

How to recreate the Security defect

- Browse to – https://visiview.hillrom.com/
- Try logging in to the application with invalid password for consecutive 5 attempts.
- Account of the User will be locked, and user will not be able to login to the application.

Evidence



2.6 Directory Traversal			
Impact	High	**Risk Rating**	High
Ease of Exploit	Easy		
Likelihood	Medium		
Category	Exposure of Information Through Directory Listing		

URL/Impacted system	https://visiview.hillrom.com/static

Description

Directory traversal occurs when an attacker obtains unauthorized access to the contents of a directory or file on the server by exploiting how the server dynamically generates paths to those resources. There are traditionally two ways to refer to resources on a file system:
* The absolute path to the resource
* An abstract path which contains control characters that alter the path through directory change (including, but not limited to, ../,..\, /,\, and ~/)
The application currently uses untrusted data to construct the path and filename of a given resource, which is then returned to the user. Since the application does not sanitize or otherwise validate this input, the user is able to supply an abstract path which may refer to an unauthorized resource on the server.

Impact

Directory traversal results in an attacker gaining unauthorized access to the file system by manipulating input used by the application to construct a pathname.

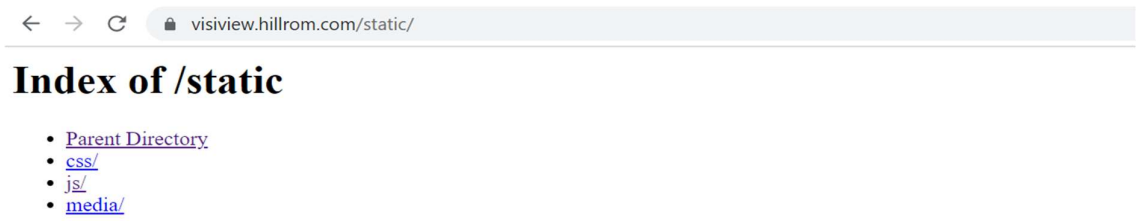
Recommendation

- Instead of using a user-supplied filename to access the file, the application should maintain a mapping from integer keys to file names, ensuring that the supplied value is both an integer and corresponds to an actual file. In the case that a value does not have a corresponding file, the application should return a generic error message. This will provide whitelist validation for the files that exist in the application.

How to recreate the Security defect

- Browse to – https://visiview.hillrom.com/static
- Notice that directories are listed

Evidence



2.7 Information Exposure leads to Account Compromise			
Impact	Medium	**Risk Rating**	Medium
Ease of Exploit	Moderate		
Likelihood	Medium		
Category	Exposure of Resource to Wrong Sphere		
URL/Impacted system	https://visiview.hillrom.com/		

Description

It is observed that multiple email addresses are disclosed in the application response. Email addresses may represent usernames of admin and other individuals that can be used at the application's login. Also observed that the application uses commonly used passwords for multiple user accounts which allow attacker to gain access to privilege user accounts.

Impact

Attacker can use the sensitive information disclosed of accounts with higher privileges and functionality misuse or compromise accounts

Recommendation

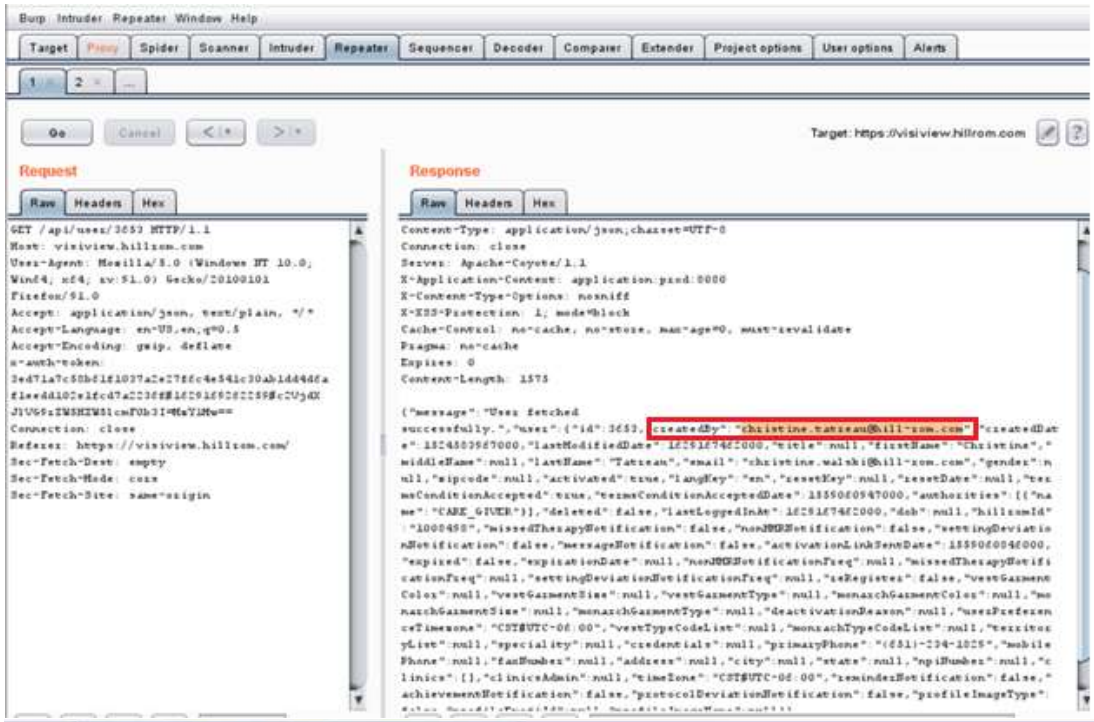
- MFA should be implemented for admin accounts.
- Proper session validation should be implemented and information specific to users should only be disclosed using proper access control mechanism.

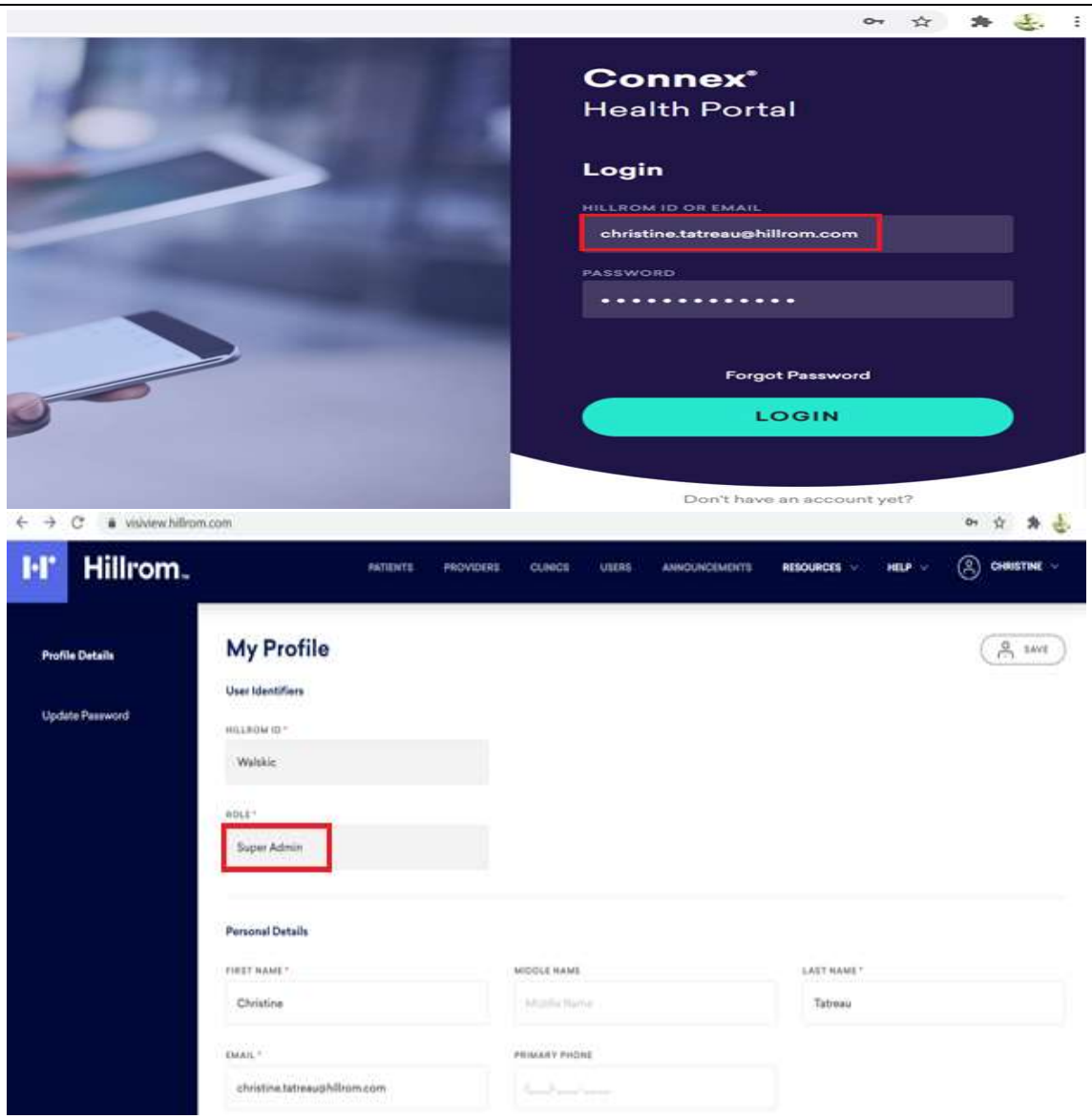
How to recreate the Security defect

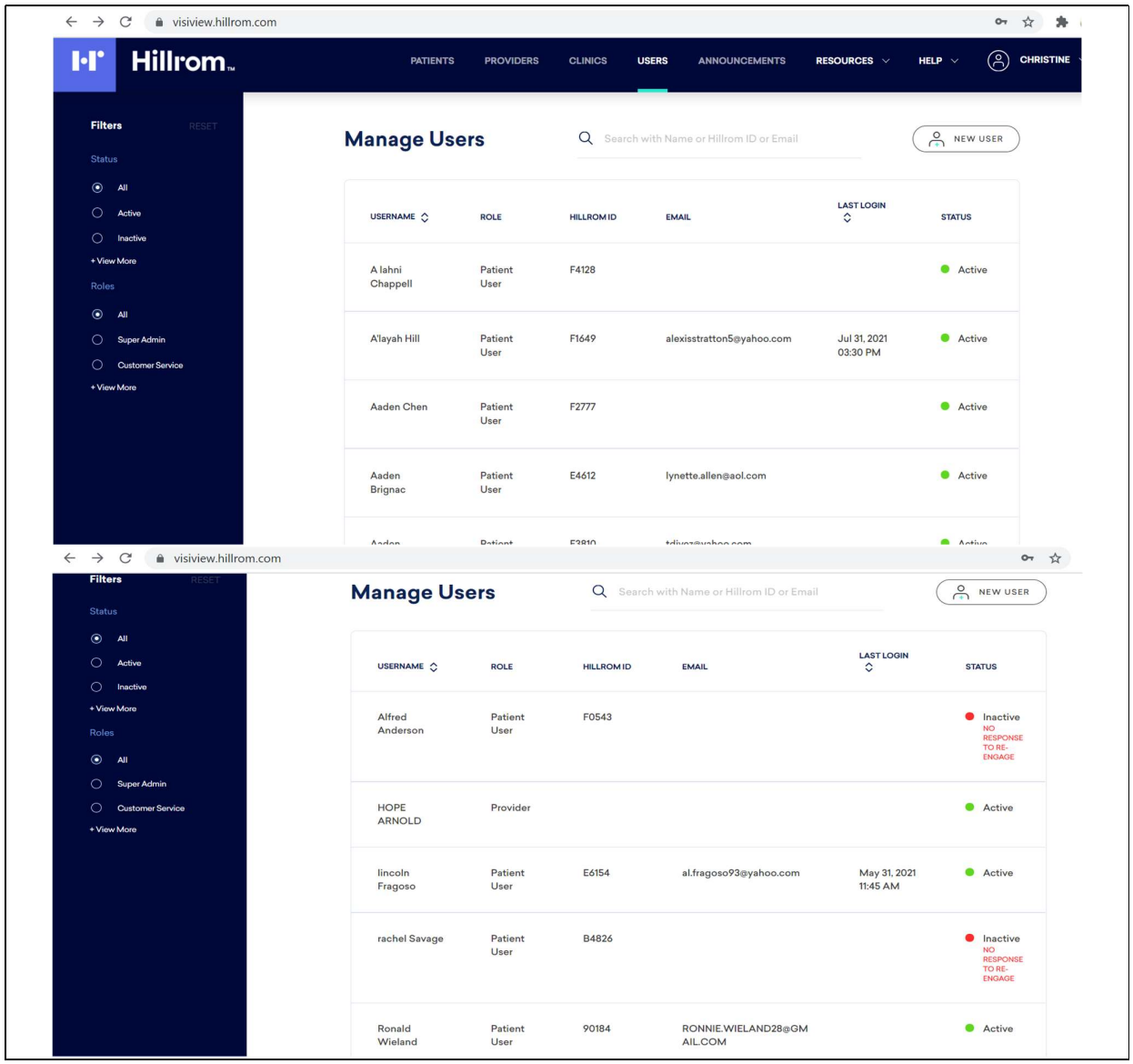
- Login to the web application– http://visiview.hillrom.com/

- Capture the traffic using burp
- Check the application response containing the email id of ADMIN user-- christine.tatreau@hill-rom.com
- Login with Email-id: christine.tatreau@hillrom.com and the default password: Myspring@2018

Evidence

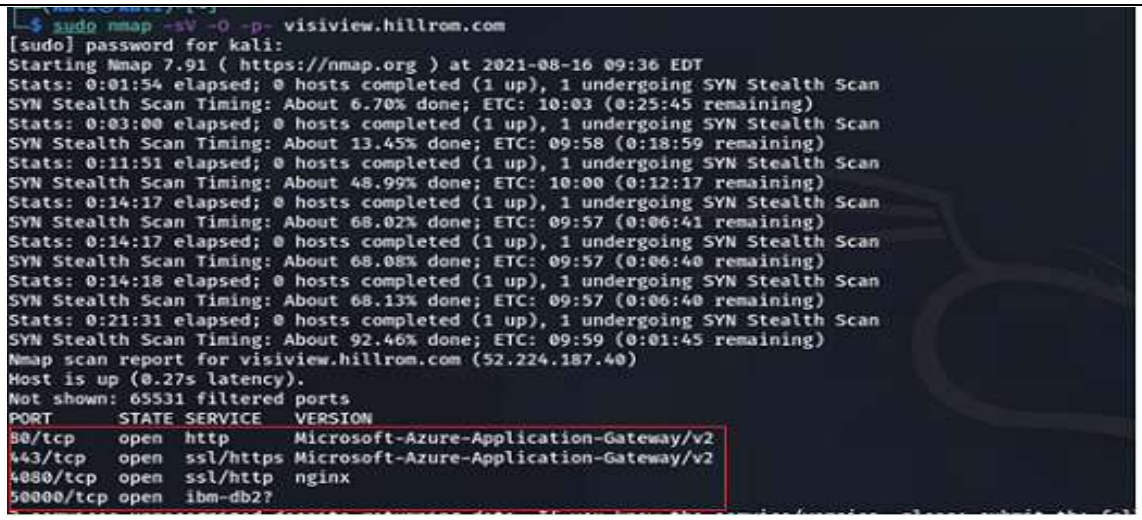



Connex®
Health Portal
Login
HILLROM ID OR EMAIL
christine.tatreau@hillrom.com
PASSWORD
Forgot Password
LOGIN
Don't have an account yet?
visiview.hillrom.com
Hillrom
PATIENTS
PROVIDERS
CLINICS
USERS
ANNOUNCEMENTS
RESOURCES
HELP
CHRISTINE
Profile Details
Update Password
My Profile
SAVE
User Identifiers
HILLROM ID *
Welskic
ROLE *
Super Admin
Personal Details
FIRST NAME *
Christine
MIDDLE NAME
LAST NAME *
Tatreau
EMAIL *
christine.tatreau@hillrom.com
PRIMARY PHONE



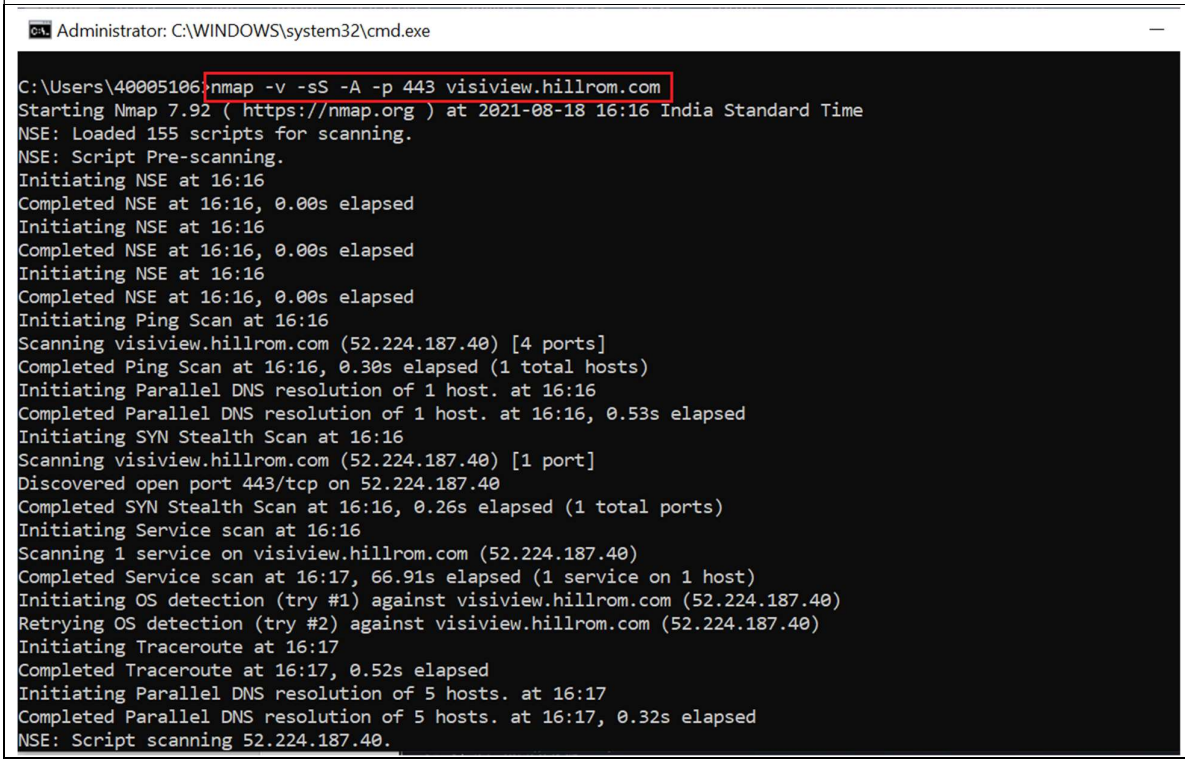
2.8 Information Exposure of Ports and Services			
Impact	Medium	**Risk Rating**	Medium
Ease of Exploit	Moderate		
Likelihood	Medium		
Category	Security Misconfiguration		
URL/Impacted system	https://visiview.hillrom.com/		
Description			
Security misconfiguration can happen at any level of an application stack, including the network services, web server, application server, database, frameworks, or storage. It is observed that the application displays open ports and servers while performing manual penetration testing. Such flaws frequently give attackers unauthorized access to some system data or functionality.			

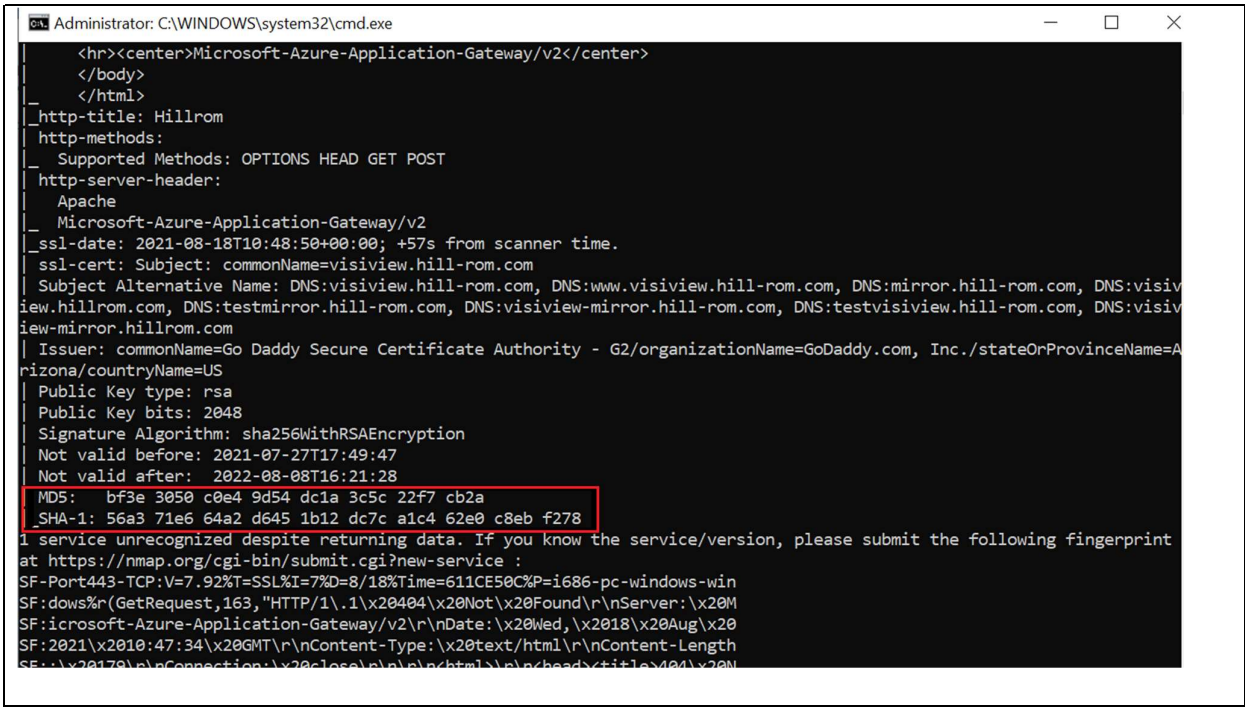
Impact
If attacker intercept network and get user's credentials that may lead user's account get compromised.
Recommendation
Configure all external and internal network devices and services supporting sensitive information to utilize an encrypted protocol for authentication and transmission of data. Configure services and devices to encrypt their network traffic with SSL/TLS, SSH, or IPSEC. If the service natively supports some form of traffic encryption, enable that feature of the service. A common example of encrypting basic network service traffic is the use of HTTPS (SSL/TLS) to encrypt standard HTTP traffic.
How to recreate the Security defect
• Run application with Kali Linux • Scan with NMAP • Found open ports.
Evidence



2.9 Weak Hashing Algorithms			
Impact	Medium	**Risk Rating**	Medium
Ease of Exploit	Easy		
Likelihood	Medium		
Category	Encryption and Authentication		
URL/Impacted system	https://visiview.hillrom.com/		
Description			
Weak hashing algorithms such as MD2, MD5 and SHA-1 are known to be susceptible to collision attacks. Collisions in hashing algorithms occur when multiple data sets can be specifically constructed to produce the same resulting digest value.			

Impact
Use of weak hashing algorithms may result in sensitive data exposure, key leakage, broken authentication, insecure session, and spoofing attack.
Recommendation
• Employ one of the SHA-2 or SHA-3 cryptographic hash functions whenever it is important so that an attacker will not be able to generate multiple pieces of data that have the same hash. Even though no practical attack against SHA-1 exists yet, it is advisable to stop using it wherever possible to avoid future problems.
How to recreate the Security defect
• Use Kali Linux/Ubuntu and run the Nmap scan on port 443
Evidence

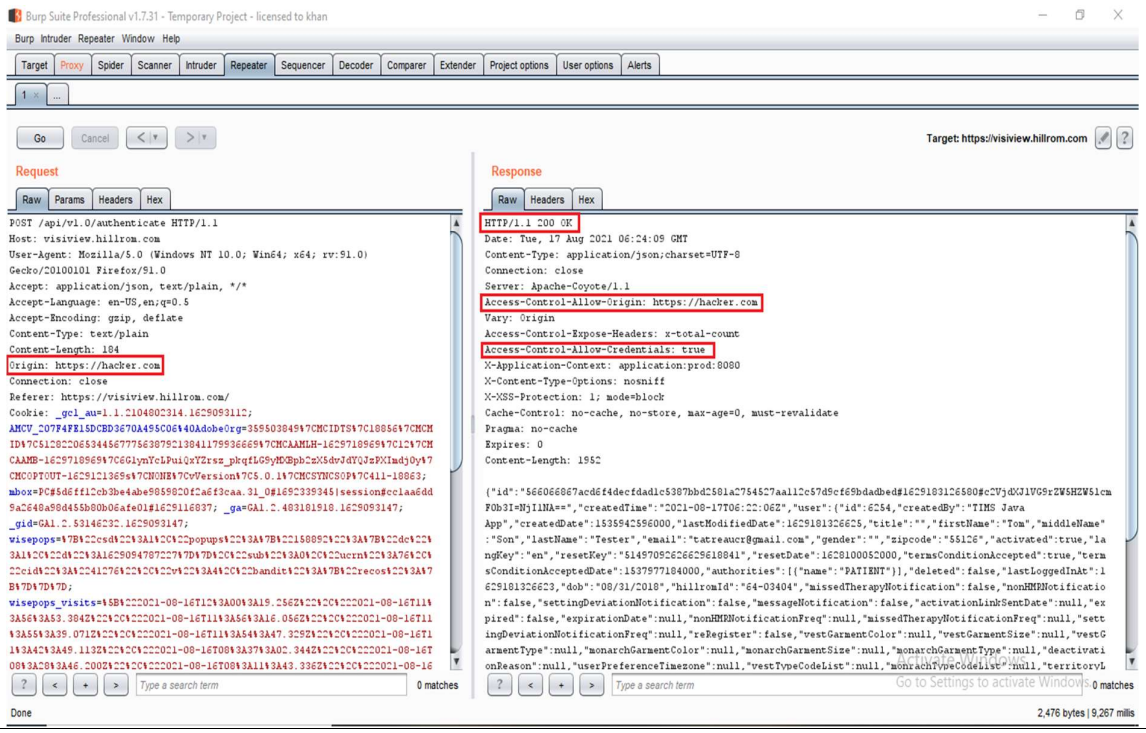




2.10 CORS Misconfiguration			
Impact	Medium	**Risk Rating**	Medium
Ease of Exploit	Moderate		
Likelihood	Medium		
Category	Permissive Cross-domain Policy with Untrusted Domains		
URL/Impacted system	https://visiview.hillrom.com/		
Description			
Access-allow control attribute is incorrectly set using wildcards such as (*) under which domains can request resources. It enables controlled access to resources which are outside of the given domain. It adds flexibility to Same Origin Policy (SOP)			
Impact			
• This is usually set as default, which means any domain can access resources on this site. Able to steal confidential and sensitive information.			
Recommendation			
• Set the Access-allow control header to validated and whitelisted websites • To implement CORS securely, you need to associate a validation list with Access-Control-Allow-Origin that identifies which specific domains can access resources. Then your application can validate against this list when a domain requests access.			
How to recreate the Security defect			
• Browse to - https://visiview.hillrom.com/ • Capture traffic observe response using burp suite			

- Change origin in request to any URL and observe response

Evidence



2.11 Click Jacking			
Impact	Medium	**Risk Rating**	Medium
Ease of Exploit	Moderate		
Likelihood	Medium		
Category	Information Disclosure		
URL/Impacted system	https://visiview.hillrom.com/		

Description

Clickjacking is a malicious technique that consists of deceiving a web user into interacting by clicking with something different to what the user believes they are interacting with. This type of attack, that can be used alone or in combination with other attacks, could potentially send unauthorized commands or reveal confidential information while the victim is interacting with seemingly harmless web pages. Visiview Hillrom has many instances where pages are missing X-frame headers to avoid clickjacking.

Impact

- Attacker loads frame with high opacity onto the victim user's application page, something which is not the same what the user believed to be interacting with.
- Proof of clickjacking instance recorded is attached in the Evidence.

Recommendation

- The X-Frame-Options HTTP response header can be used to indicate whether a browser should be allowed to render a page in a <frame>, <iframe> or <object>.
- Sites can use this to avoid clickjacking attacks, by ensuring that their content is not embedded into other sites.

How to recreate the Security defect

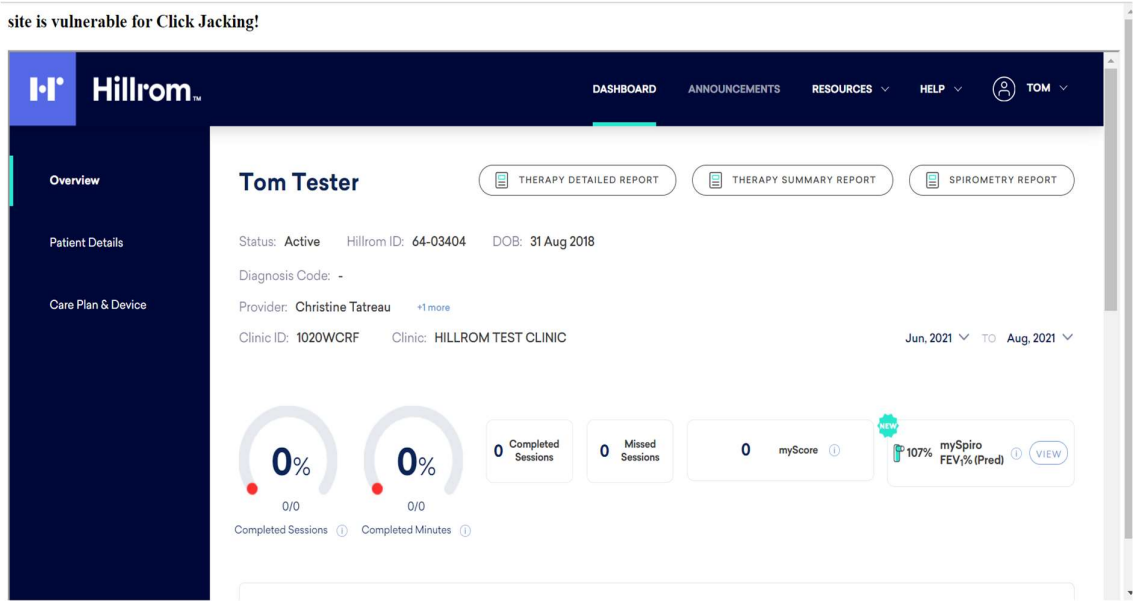
- Write the following code into a notepad and save it as clickjacking.html
- Open that in the browser

Evidence

clickjacking - Notepad

File Edit Format View Help

```
<html><head>
<title> leaseweb </title>
<style>
frame {
opacity: 0.5;
border: none;
position: absolute;
top: 0px;
left: 0px;
z-index: 1000;
}
</style>
</head>
<body>
<script>
window.onbeforeunload = function()
{
return " Do you want to leave ?";
}
</script>
<p> <h3>site is vulnerable for Click Jacking!<h3></p>
<iframe id="frame" width="100%" height="100%"  src="https://visiview.hillrom.com"></iframe>
</body>
</html>
```



2.12 Weak Change Password Policy			
Impact	Medium	**Risk Rating**	Medium
Ease of Exploit	Easy		
Likelihood	Medium		
Category	Unverified Password Change.		
URL/Impacted system	https://visiview.hillrom.com/		

Description

During the manual penetration testing of application, it was observed that the application's authenticated password reset mechanism uses the existing password or current password to execute a password change. Accepting the previous password, is a bad policy. The reason of change the passwords is security, if the old password still works; there's no point of changing it. So, make sure that the old password is useless.

Impact

- If the old password was revealed to someone else (can happen, especially with phishing) then there is a chance to login with the credentials and can gain access to the application.
- It is possible someone may gain access to your saved passwords.

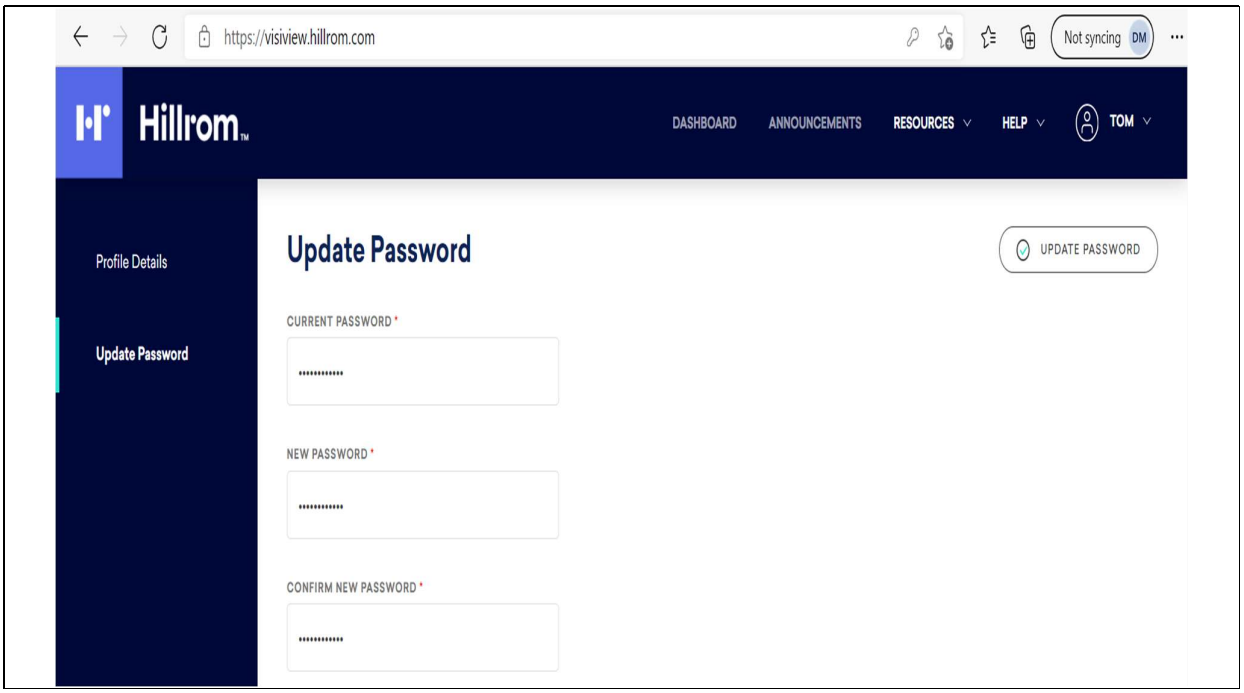
Recommendation

- Changing a user's password after a fixed amount of time is requested to keep the account reasonably safe if the authentication credentials are stolen or leaked. Hence, accepting previous passwords is an unsafe solution and should be avoided**.**
- Introduce additional authentication controls (i.e., multi-factor authentication).
- "Enforce password history" option is used to prevent users from reusing old passwords. This makes the system more secure; a user needs to use a new password (one that has never been used before) each time they change the password.

How to recreate the Security defect

- Browse to – https://visiview.hillrom.com/
- Navigate to Profile→ Update password
- Enter the new password same as the Current password.
- Notice that the application accepts the password.

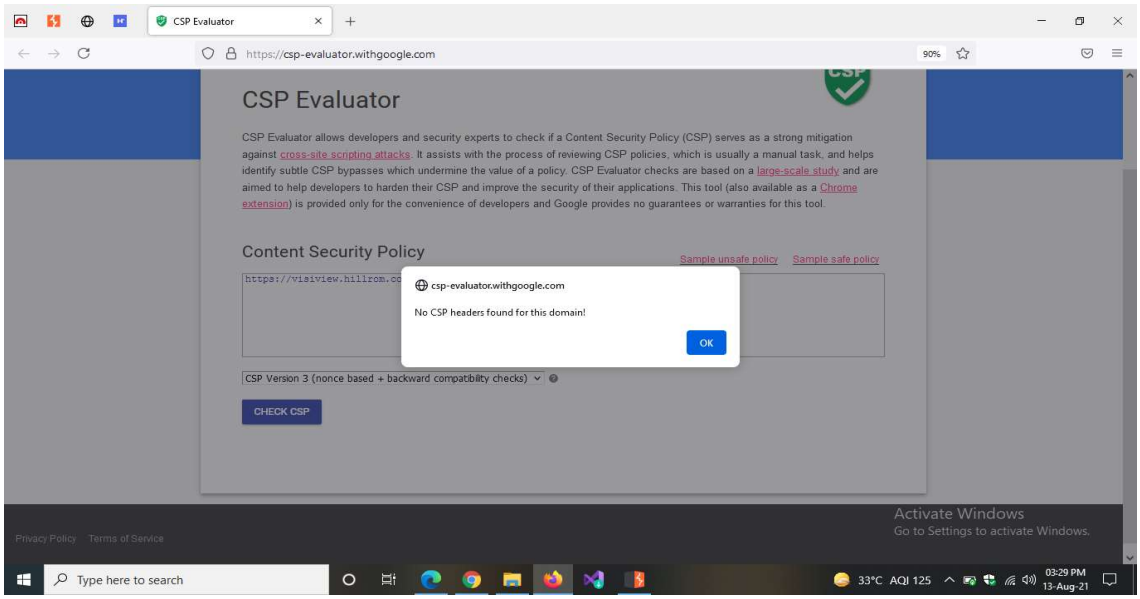
Evidence



2.13 No Content Security Policy			
Impact	Low	**Risk Rating**	Low
Ease of Exploit	Difficult		
Likelihood	Medium		
Category	Improper Restriction of Rendered UI Layers or Frames		
URL/Impacted system	https://visiview.hillrom.com/		
Description			
During the HTTP traffic analysis of the Visiview Hillrom web interface, it was observed that the server does not support Content Security Policy. Content Security Policy is a standard that helps protect against various content injection attacks like cross site scripting. While the victim is interacting with seemingly harmless web pages.			
Impact			
Without a Content Security Policy, an attacker can perform content injection attacks if data from the service is displayed in a browser			
Recommendation			
Enabling Content Security Policy response header to all HTTP server responses helps in preventing content injection attacks. While adding Content Security Policy it must be set correctly specifying the locations from which content can be loaded. Content-Security-Policy: <Policy-directive>;			
How to recreate the Security defect			
• Browse to - https://csp-evaluator.withgoogle.com/ • Enter the URL - https://visiview.hillrom.com/			

- Click on check CSP

Evidence



2.14 Text injection on Visiview			
Impact	Medium	**Risk Rating**	Low
Ease of Exploit	Difficult		
Likelihood	Low		
Category	User Interface (UI) Misrepresentation of Critical Information		
URL/Impacted system	https://visiview.hillrom.com/		

Description

Text injection is a basically injection in which user input is reflected as it is in the application response as plaintext. This is one of the ways to perform content spoofing or virtual defacement which can be used in phishing attacks.

Impact

An attacker can use text injection vulnerability to present a customized message on the application that can phish users into believing that the message is legitimate. The intent is typical to tick victims, although sometimes the actual purpose may be to simply misrepresent the organization or an individual.

Recommendation

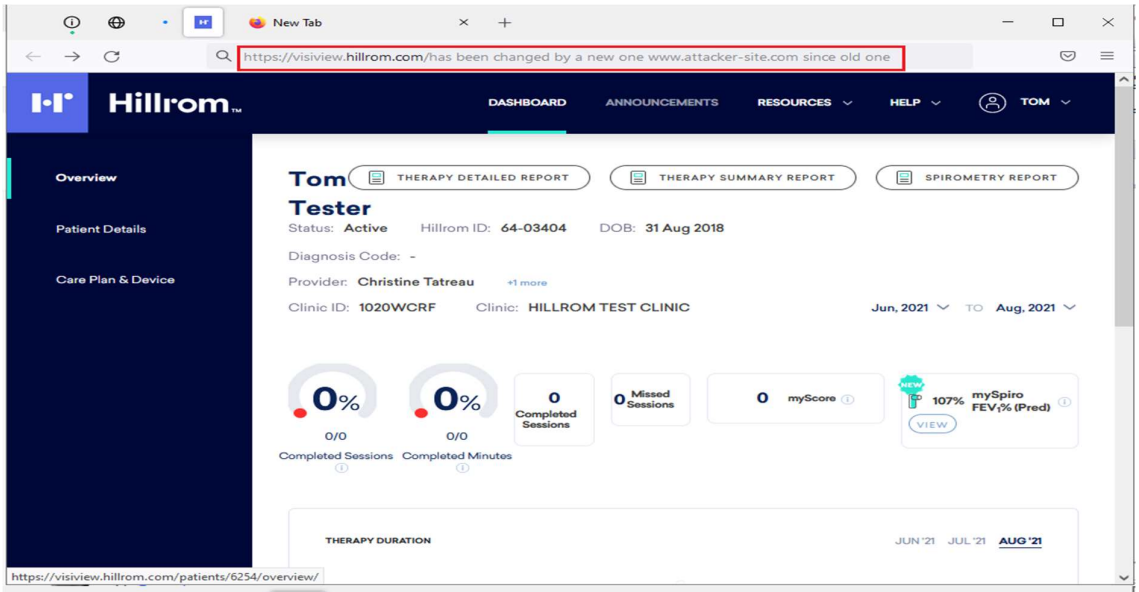
- The application should only accept the values and types that are defined for parameters and should be checked at the server-side whether there is change content, if there is change, then the application should reject that request.

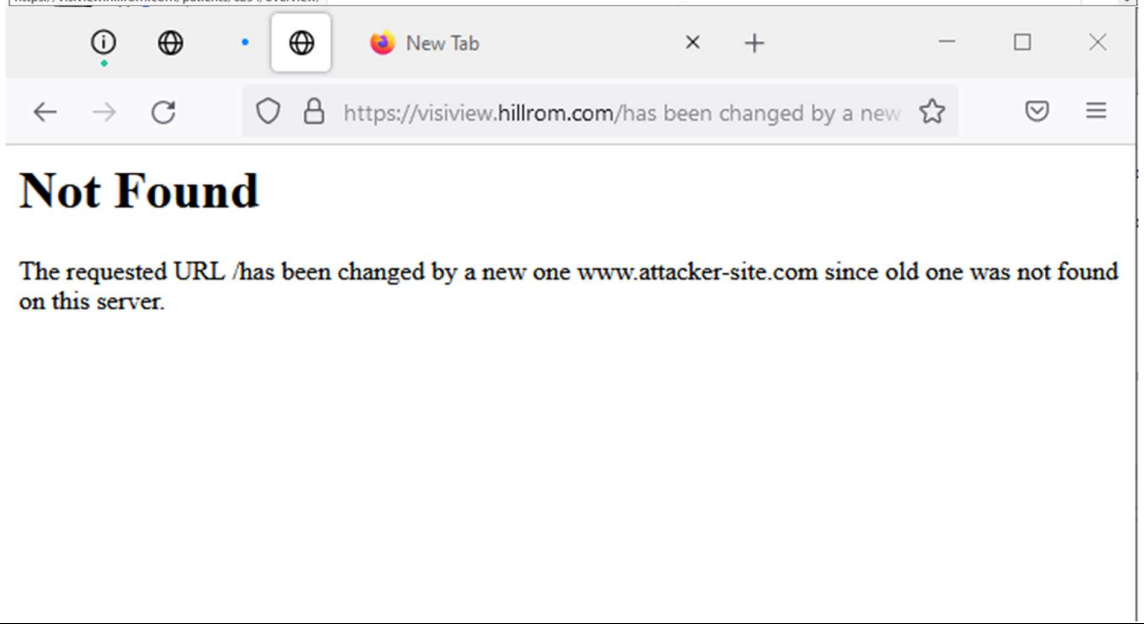
- Never construct and send messages via URL in the page response. Prefer using messages predefined in a property file.

How to recreate the Security defect

- Browse to – https://visiview.hillrom.com/
- Add content-type: text/plain in the URL
- Application is giving default error pages

Evidence





2.15 No Multifactor Authentication (MFA)			
Impact	Medium	**Risk Rating**	Low
Ease of Exploit	Moderate		
Likelihood	Low		
Category	Use of Single-factor Authentication		
URL/Impacted system	https://visiview.hill-rom.com/		

Description

The application uses single-factor authentication to authenticate privileged users to the system. Single-factor authentication refers to the use of a single component to identify an end user to an application or system. The factor provided may be something the user knows, something the user is, or something the user has. Each of these options provides their own set of advantages and risks when used for authentication:

- "Something you know", such as a user-defined password, may be easily created and changed when necessary. Authentication factors derived from the end user must have some degree to be managed by the user themselves, leaving the known secret's security up to them. This can result in the secret being forgotten or exposed through a breach of a separate system that holds or uses the same known secret.
- "Something you are", such as a fingerprint, provides an end user with a constant factor that cannot be easily acquired or mimicked by an attacker. While this initially provides a strong barrier to entry and will always be with the end user, a single breach could leave the attribute used for authentication useless as it cannot be updated.
- "Something you have", such as a hardware token, can be managed from a central source and is configured to constantly update, removing responsibility for the known secret from the user. However, this transition of the knowledge base may hinder the application's accessibility if the device is not always at hand.

Impact

If an attacker compromises the authentication mechanism (e.g., a victim's account password), they will have full access to functionality and data normally only available to the victim.

Recommendation

Multi-factor authentication should be implemented and enforced for externally accessible applications containing sensitive data or functionality. Multi-factor authentication is built upon the combination of two or more components that can prove a user's identity to the application. This provides an additional layer of security as it is assumed that an unauthorized attacker will not be able to supply both factors required for authentication. The factors required by the application should be a combination of at least two distinct factors from the following:

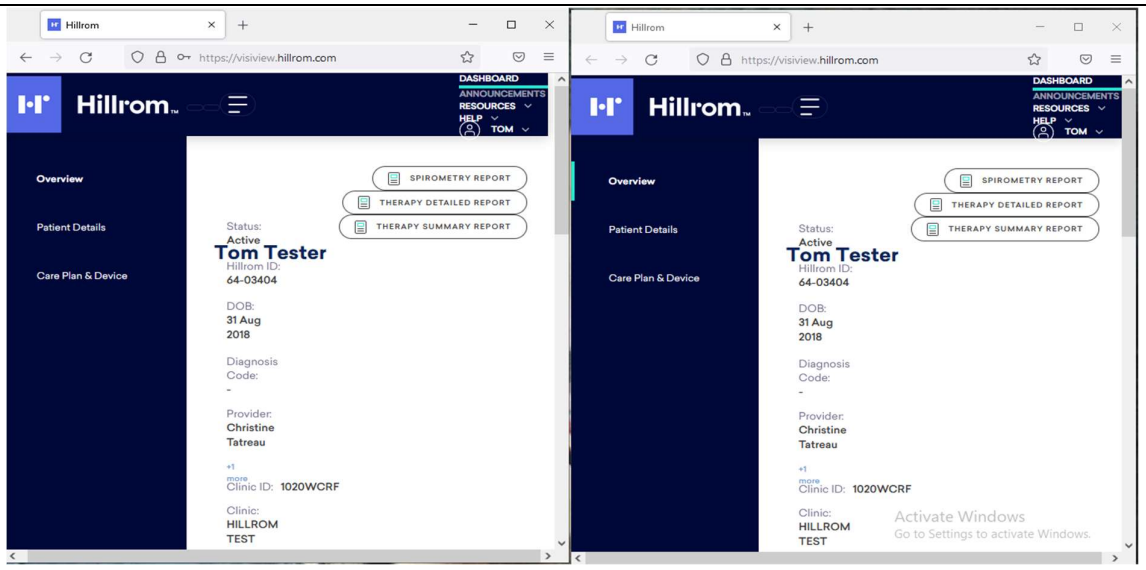
- Something the user knows
- Something the user is
- Something the user has

For example, a common multi-factor authentication mechanism requires a user to provide a password they have created (something they know), as well as a value from a hardware token (something they have). If an attacker can compromise a user's password, they will still not have access to the hardware token and will not be able to gain access to the system.

Note: requiring two or more pieces of information for authentication that fall under the same factor category does not provide true multi-factor authentication. For example, a user's password

and the answer to their security question are both something the user knows. Requiring both during authentication does not represent true multi-factor authentication.
How to recreate the Security defect
• Browse application • Check the application login without MFA

2.16 Application Allows Concurrent Sessions			
Impact	Medium	**Risk Rating**	Low
Ease of Exploit	Difficult		
Likelihood	Low		
Category	Manage User Sessions		
URL/Impacted system	https://visiview.hill-rom.com/		
Description			
The application allows concurrent sessions; multiple users can login to the application simultaneously with the same user credentials.			
Impact			
Attacker can make victim's account active with the same username and password.			
Recommendation			
The application should only allow a user to establish a single session with a particular set of credentials at a time. Once that session has been established, subsequent attempts to login using those credentials should either be denied or existing sessions should be terminated, depending on business needs. If concurrent sessions are required for business purposes, additional session management features must be provided to ensure that all end users are made aware of multiple sessions. Such features include allowing end users to view all current sessions, prompting users when a new session is created, and providing users the ability to terminate unwanted sessions. Additionally, when allowing concurrent sessions, it is recommended that users are notified that their credentials were used to establish a new session, including the time and IP from which the session was established			
How to recreate the Security defect			
• Login into the application • Capture traffic observe response • Again, Login in the new tab with same credential as before • Notice that we multiple users can login to the application simultaneously with the same user credentials.			
Evidence			



2.17 Exposed Subdomains			
Impact	Low	**Risk Rating**	Informational
Ease of Exploit	Medium		
Likelihood	Low		
Category	Configuration		
URL/Impacted system	https://visiview.hillrom.com/		

Description

During the assessment, various subdomains were detected on target domain names that are accessible to the public Internet and it uses insecure HTTP connection as well.

Impact

- There is a possibility that this subdomain may be a portal to administrative functionalities for various enterprise applications.
- Possibility of subdomain takeover

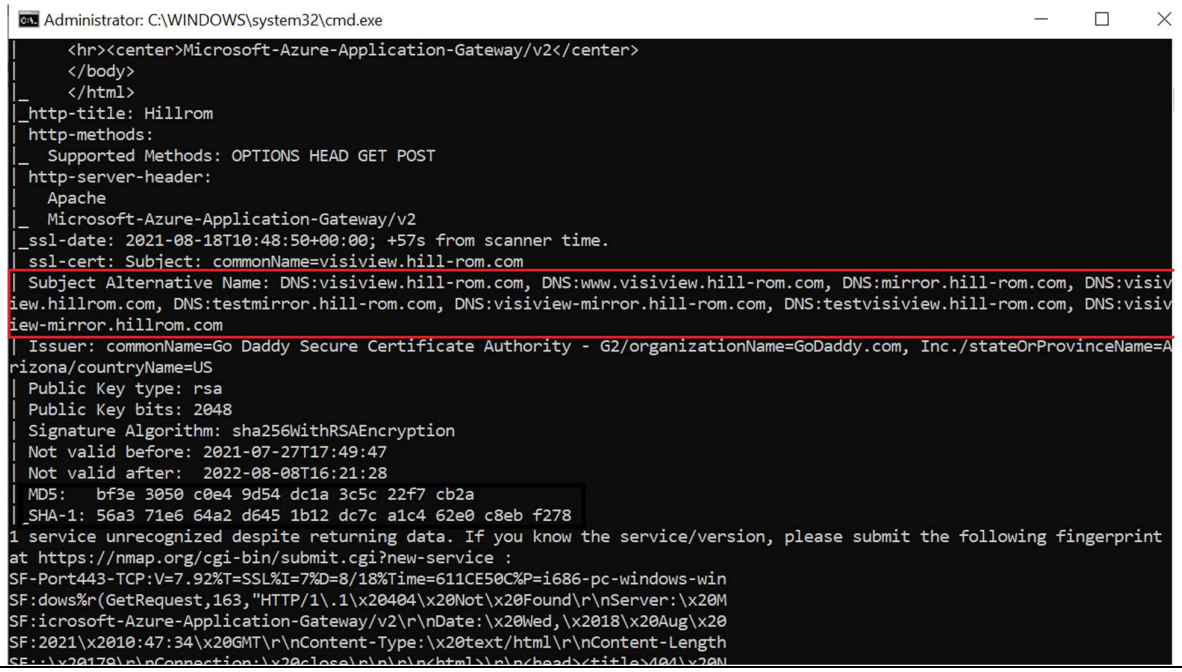
Recommendation

- Remove public access to the subdomains
- Remove the unnecessary subdomain

How to recreate the Security defect

- Browse application
- Capture traffic observe response
- Found the response displaying the Apache tomcat version

Evidence



2.18 HTTP Strict Transport Security Not Enabled			
Impact	Low	**Risk Rating**	Informational
Ease of Exploit	Medium		
Likelihood	Low		
Category	Cleartext Transmission of Sensitive Information		
URL/Impacted system	https://visiview.hillrom.com/		

Description

During Assessment, it was observed that HSTS is not enabled for web application. While the application may initially be served over HTTPS, it is also accessible over HTTP, resulting in application traffic being sent in plaintext.

Impact

- Attacker can perform Man in the middle attack also can see all communication in clear text

Recommendation

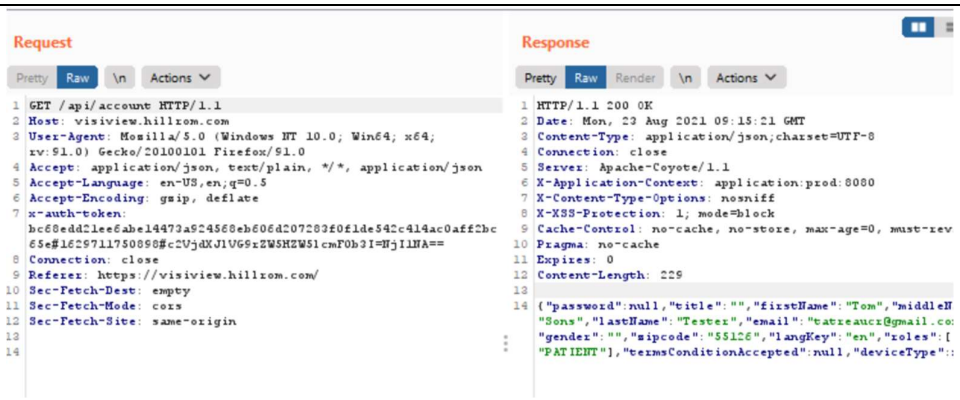
- HTTPS should be enabled and enforced on the application server. Once HTTPS has been properly configured, ensure that the application requires HTTPS for access to all application resources, including JavaScript files, style sheets, and images. When a user attempts to navigate to any part of the application over HTTP, the application should redirect the user to the HTTPS version of the application. Lastly, it is recommended that applications are deployed with HTTP Strict Transport Security (HSTS). HSTS forces the browser to access a site only over HTTPS, and prevents access in cases where the authenticity of the X.509

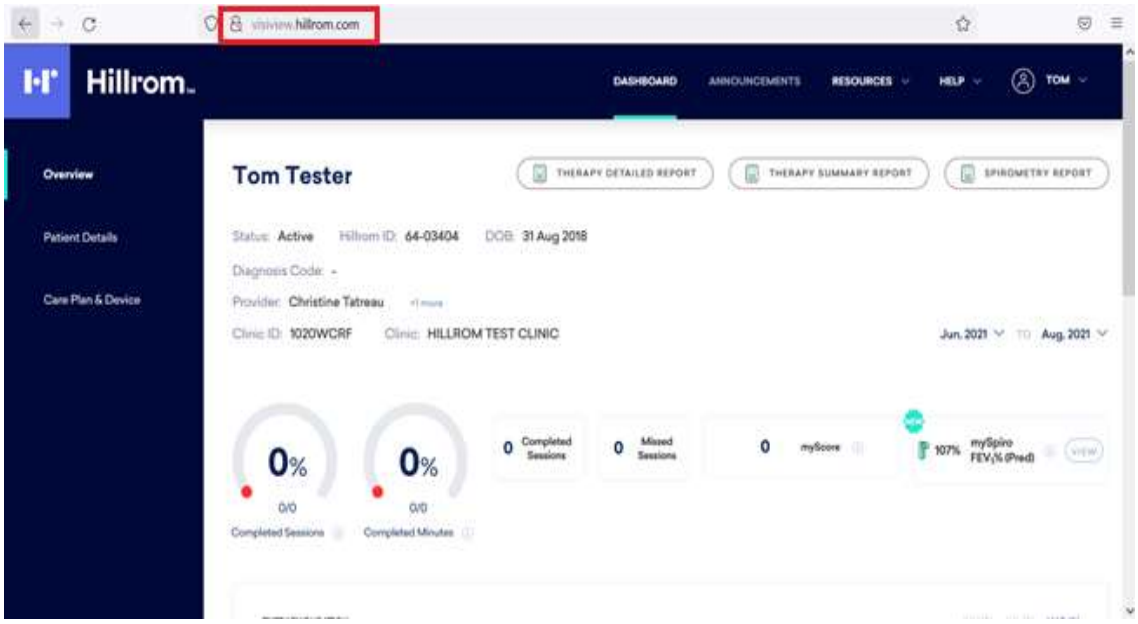
certificate cannot be verified. HSTS is supported in recent versions of the Chrome, Firefox, and Opera web browsers. To enable HSTS, simply add the Strict-Transport-Security header to the response header when users first access the site over HTTPS. For more information on the HSTS header, refer to the relevant OWASP page, located at https://www.owasp.org/index.php/HTTP_Strict_Transport_Security.

How to recreate the Security defect

- Browse application https://visiview.hillrom.com/
- Capture the traffic and check for the header

Evidence





2.19 Server Banner Disclosure			
Impact	Low	**Risk Rating**	Informational
Ease of Exploit	Difficult		
Likelihood	Low		
Category	Information Exposure		
URL/Impacted system	https://visiview.hillrom.com/		

Description

While performing vulnerability assessment and penetration testing, it was observed that verbose server information of Visiview Hillrom is sent in the HTTP responses from the server. The information is commonly included in the server response headers and can disclose information like server name, type, and version number.

Impact

Verbose server banners provide additional information that allows an attacker to perform targeted attacks to the specific technology stack in use by the application and underlying infrastructure.

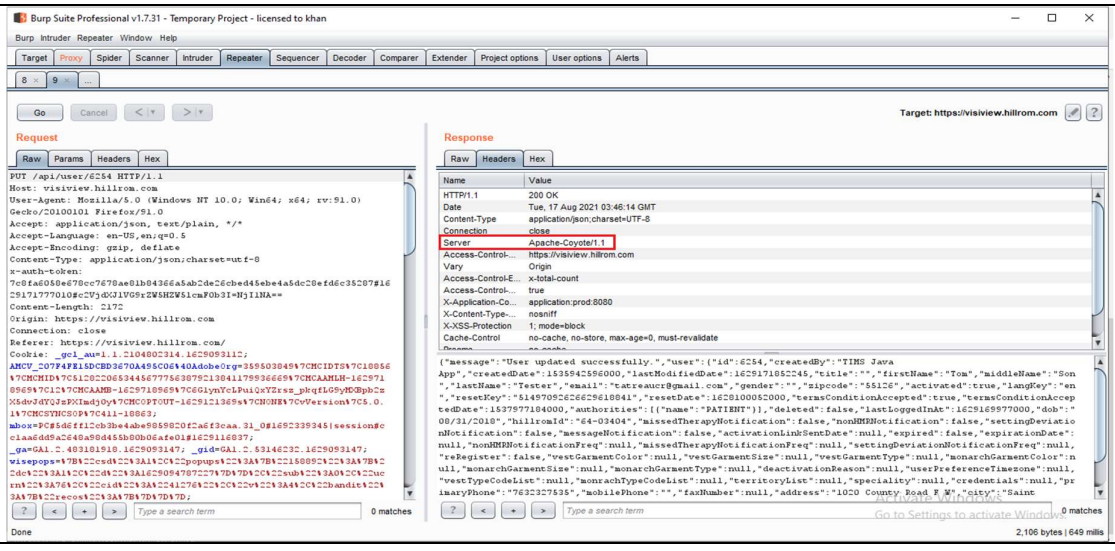
Recommendation

- Verbose server information should be removed from all HTTP responses. This can be performed by modifying the server's configuration files or through the use and configuration of a web application firewall.
- It is recommended to use generic error message response from server, so that server banner is disclosed in the error message response from the server.

How to recreate the Security defect

- Browse to - https://visiview.hillrom.com/
- Capture the request in burp
- Found server name in response

Evidence



2.20 Cross-site Scripting (XSS) on Respiratorycare			
Impact	Low	**Risk Rating**	Informational
Ease of Exploit	Difficult		
Likelihood	Low		
Category	Reflected XSS into HTML context with nothing encoded		
URL/Impacted system	https://respiratorycare.hill-rom.com/		

Description

It was observed that web application's 'Resources' page--https://respiratorycare.hill-rom.com/ is vulnerable to cross Site Scripting (XSS). Cross-site scripting is a web security vulnerability that allows an attacker to compromise the interactions that users have with a vulnerable application to access any of the user's data. If the victim user has privileged access within the application, then the attacker might be able to gain full control over all the application's functionality and data.

Impact

If the compromised user has elevated privileges within the application, then the impact will generally be critical, allowing the attacker to take full control of the vulnerable application and compromise all users and their data.

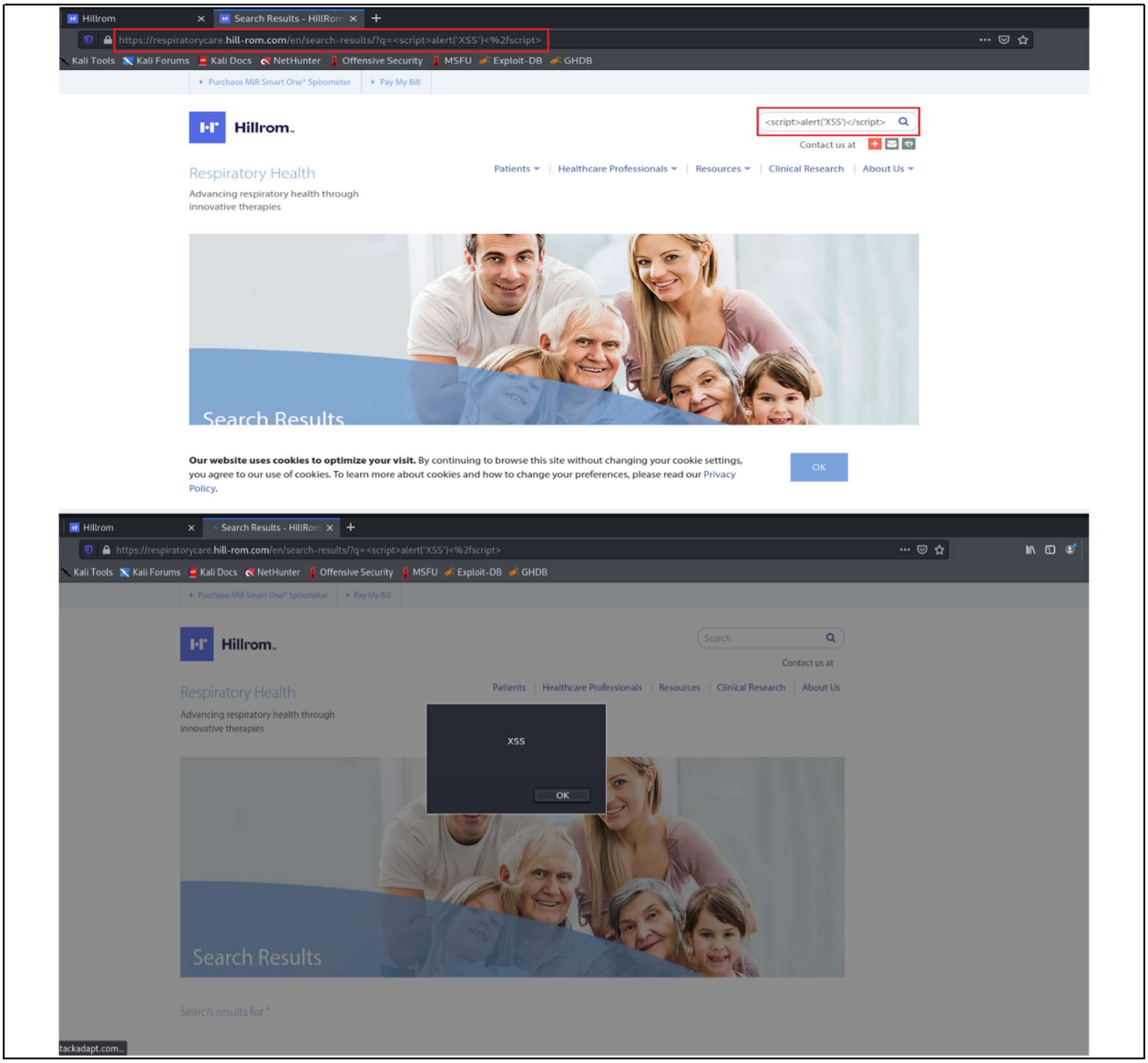
Recommendation

- Validate to catch potentially malicious user-provided input
- Encode output to prevent potentially malicious user-provided data from triggering automatic load-and-execute behaviour by a browser
- Use Content Security Policy (CSP) to reduce the severity of any XSS vulnerabilities that still occur.

How to recreate the Security defect

- Browse to – https://visiview.hillrom.com/
- Navigate to Resources and select any one option
- Perform a cross-site scripting attack that calls the `alert` function in the search field

Evidence


Hillrom
Search Results - HillRom
https://respiratorycare.hill-rom.com/en/search-results/?q=<script>alert('XSS')<%2fscript>
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
MSFU
Exploit-DB
GHDB
Purchase MIR Smart One® Spirometer
Pay My Bill
Hillrom.
<script>alert('XSS')</script>
Contact us at
Respiratory Health
Advancing respiratory health through innovative therapies
Patients
Healthcare Professionals
Resources
Clinical Research
About Us
Search Results
Our website uses cookies to optimize your visit. By continuing to browse this site without changing your cookie settings, you agree to our use of cookies. To learn more about cookies and how to change your preferences, please read our Privacy Policy.
OK
Search
XSS
OK
Search results for "
tackadapt.com...

3. Abbreviation

APP	Application
HTML	Hyper Text Mark-up Language
HTTP(S)	Hypertext transfer protocol (Secured)
Pg.	Page
TLS	Transport Layer Security
SSL	Secure Sockets Layer
IP	Internet Protocol
LTTS	Larsen & Toubro Technology Services
SOP	Same Origin Policy
OWASP	Open Web Application Security Project
VAPT	Vulnerability Assessment and Penetration testing
IDOR	Insecure direct object references
SOP	Same Origin Policy
MFA	Multifactor Authentication
CSP	Content Security Policy
CORS	Cross-origin resource sharing
XXE	XML External Entities
XSS	Cross-site Scripting

4. Appendix
Vulnerability scan reports.